Massachusetts Institute of Technology

Department of Electrical Engineering and Computer Science

Proposal for Thesis Research in Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy

Title: Role transfer for robot tasking

Submitted by: Paul Fitzpatrick
NE43-936
200 Technology Square
Cambridge, MA 02139

_____
(Signature of author)

Date of submission: April ??, 2002

Expected Date of Completion: April 2003

Laboratory where thesis will be done: MIT Artificial Intelligence Laboratory

Brief Statement of the Problem:

Robotics has proven most successful in narrowly defined domains that offer sufficient constraints to make automated perception and action tractable. The goal of this thesis is to take a step towards generality by developing methods for applying a robot to many different narrow domains. This is complementary to the more common research goal of enhancing machine perception and action to deal with wider domains. This approach to extending the range of application of a technology through parameterization rather than generalization is key to fields such as automatic speech recognition. It has the theoretical advantage of providing a framework for factoring context into perception, and the practical advantage of creating systems that do useful work with limited technology.

I propose a scheme for communicating constraints to a mechanically general-purpose robot, so that it can perform novel tasks without needing to first solve open problems in perception and action. In particular, this thesis develops mechanisms for communicating the structure of simple tasks to a robot, translating this structure into a set of supervised learning problems for parts of the task which are difficult to communicate directly, and solving those problems with the guidance of a protocol for inducing feature selection.

1 Introduction

Robots are designed to interact tightly with the physical world. To build working systems using today's technology, the complexity of this interaction must be minimized by carefully engineering the environment and/or the task. But if robots are to become more generally useful, then we need to find ways for them to operate amidst poorly characterized real-world objects and processes. I propose working towards the ideal of unconstrained domains by exploring a precursor: the domain of robot tasking.

Robot tasking addresses how to induce a mechanically (more or less) general-purpose robot to perform a particular operation. A common goal is for the "inducement" to be consistent with natural human behavior, which can be taken as a de facto standard for communicating about physical objects and actions. For the purposes of this thesis, the crucial objective of tasking is to acquire a sufficient characterization of the world from a human tutor to provide the context that machine perception needs.

Perception in context

All perception is indirect, accumulating measurements of observable features to estimate the hidden state we care about. When a robot is designed to work within the context of a single well-defined task, it can make use of measurements that happen to correlate well with the hidden states under those circumstances, but which could not be relied upon under less constrained conditions. For example, a robot arm operating on a conveyor-belt may estimate the size of an object by measuring the number of above-threshold pixels in its silhouette. This measurement implicitly depends on good background contrast, stable viewing distance, and consistent object pose – none of which hold in general. Under other circumstances, perhaps these particular measurements would no longer work, but another set might. It is possible to imagine that we could cover a wide range of tasks with locally valid estimation methods, without ever developing a general estimation technique that works in all of them. Of course, if the robot is expected to be autonomous, this does us little good since we need to know which set of measurements under which circumstances. But in robot tasking, where a human is available to actually set the task, we have an independent source of context.

Of course, this is no substitute for improving perceptual abilities, but it does allow us to get further with what we already have. This form of generality by parameterization is a recurring pattern in many fields where truly general solutions are currently infeasible. For example, the field of automatic speech recognition had developed general techniques for creating speech recognition systems tailored to particular domains.

Communicating context

At one level, the difference between tasking a robot and interacting with a computer seems merely a question of interface modality – voice and gesture versus keyboard and mouse, for example. In fact, the non-computational nature of the robot's environment fundamentally alters the nature of possible interfaces. For example, the critical notion of a reference/pointer/link/binding becomes a very different beast when the objects referred to are outside computational control. Robot tasking provides a very pragmatic, task-oriented domain in which to address this problem of reference in its various guises: the "Symbol Grounding Problem", the "Binding Problem", and assorted "Correspondence Problem"s. This thesis starts from the observation that within the context of a sufficiently constrained task, none of these Problems seem particularly Problematic. This suggests that for interface purposes, perhaps task structure needs to be made central, with references being derived constructs rather than atomic. I motivate this possibility and explore it further in section 2. Briefly, my approach is to allow a human tutor to first communicate the structure of the task to the robot, and to use the focus this model gives to incrementally ground that structure by establishing references between task state and the physical world.

Scripting

Various mixtures of programming and online training have been developed by robotics researchers. In particular, Thrun (1998) proposes a mechanism of developing control software for embedded systems. He developed a language in which "gaps" can be left in programs, where values are left to be determined by function approximators such as ANNs which are trained up by examples. The language also captures some common idioms of probabilistic reasoning.

If we look at the development of conventional computer applications, we see an evolution in the degree of control that "power users" can exercise over them. It is now common for applications that once were controlled by simple configuration files or a few "option" dialog pages to now expose very complete interfaces to allow manipulation by Turing-complete scripting languages. If we consider the requirements of robot tasking, we could imagine following a similar route as the competence and flexibility of our robots grows – in which case, it would be premature to examine these issues now. But I claim that the reverse is actually true. Since we need to get as much contextual information as we possibly can from a human tutor to make perception possible, we need flexible interfaces first – and hope that as our robots become more competent we can *reduce* the need for structured input.

This thesis essentially proposing to widen the "gaps" Thrun proposes, to the extent that they offer something analogous to the scriptable interface offered by today's computer applications. I will call this the invocation interface. At its simplest, it should allow a user to request the robot to perform a familiar task. But it should also facilitate introducing the robot to a a new task that lies within its competence range once enough perceptual context has been supplied. These two seemingly disparate requirements on the interface are intended to capture the fact that there is actually a smooth continuum between invoking a familiar task in a slightly different context and introducing a completely new task.

My hope is to develop an interface that is usable by naïve users for simple instances of task invocation. Complex task induction, on the other hand, may be more akin to programming/scripting than natural human teaching. Attempting to make this completely natural would require a long excursion through research territory that is orthogonal to the main thrust of this thesis.

Incremental role transfer

In programming, we spend time writing text, correcting it, and then putting it into operation and debugging the results. This is an inconvenient model of interaction for robot tasking for many reasons. A theoretical problem is that it requires all elements of the task description to have simple textual representations, but supporting such a unified representational scheme is one of the Hard Problems we are trying to side-step. A practical problem is that being tied to a keyboard and monitor makes it difficult to have hands and body free to demonstrate the task. But we cannot simply replace the keyboard with voice input, for example, since programming is simply not a linear process.

My approach is to cast the problem of introducing the robot to a new task as an exercise in role transfer. Initially the human takes full responsibility for the task. Step by step, the robot takes over parts of the task, with the human completing the rest so that at all times the robot is presented with the full task.

This is similar to a process seen in infant-caregiver interactions (Trevarthen 1979). It has the advantage of gracefully handling scenarios in which the human is an irreplaceable part of the task. Such cooperative tasks are likely to be very important in practice for any practical application of robot tasking, since humans will likely remain far more physically competent than robots are for some time to come.

Related Work

Cross-channel Early Lexical Learning (CELL) is a model of early human language learning implemented by Roy (1999). CELL searches for speech segments that reliably predict the occurrence of visual objects (conjunctions of shapes and colors). The model has been applied both to a corpus of infant-directed speech and to speech directed at a parrot-like robot Toco. After lexical learning, Toto could locate objects matching spoken words, and generate descriptions of objects in terms of shape and color categories. For my work, I draw guidance from the use Roy makes of cross-modal correlation. But I am not concerned with language learning; for my purposes vocalizations are to a first approximation merely transient intermediaries whose meaning need not extend beyond their use in a particular task demonstration.

Tim Oates has done work on the discovery of "language units" (Oates, Jensen & Cohen 1998, Oates 1999), including frequently occurring acoustic patterns. Some of the work I do has the same goal, but I try to achieve these ends using the complete probabilistic modeling capabilities of modern speech recognizers, including both acoustic and language models. I draw on speech recognition and understanding work done at MIT as embodied in the SUMMIT recognizer (Glass, Chang & McCandless 1996). Work on dialog systems is also relevant, particular the perceptual components of Cassell's Rea (Cassell 1989). This system is concerned with generating and recognizing natural human conversational behavior - speech and the various gestures that accompany speech such as beats.

Goldberg & Mataric (1999) has developed augmented markov models as a practical tool for modeling interaction dynamics between a robot and its environment; this form of model is well suited to the process modeling needed in role transfer. Other work at the USC Interaction Lab on robot teaching (Nicolescu & Mataric 2001, Mataric 2000) is also relevant.

Billard (2001) has demonstrated some simple mimicry and sequence learning for a robot doll and an instrumented human. The DRAMA architecture she uses is well suited to situations where the robot's state can be represented by a simple, clean vector of binary percepts. I see her work as addressing half of the problem of task learning: an ideal structure of the task can be communicated to the robot, but the mapping of that structure onto the noisy perceptual world is not communicated – rather, it is avoided by instrumenting the human until the percepts are sufficiently clean to make it a non-issue. This is precisely the problem I wish to address in this thesis.

I discuss other, more tightly related work within the individual sections that follow.

2 Establishing physical references

In conventional programming environments, all objects can be referred to in a more or less straightforward manner. This might be as simple as a pointer to a memory address, or as complex as a broker-mediated network binding to a remote resource. The huge variety of mechanisms of reference are enabled by the stability and reliability of the underlying objects.

For tasking, we need to refer to physical objects and actions. But such entities are unstable in a number of senses. First, they are poorly defined, with more or less arbitrary and fluid boundaries. Second, they are only indirectly observable through a sensing process that it would be over-generous to call "noisy".

Dealing with hidden state has been addressed in the machine learning literature, and is amenable to probabilistic treatment. It is less clear what to do about the fact that the nature of the hidden states themselves is poorly defined. For this problem, I have found it useful to look to the language acquisition literature. Language is all about reference, and the ability of infants to acquire language depends on an ability to decode such references.

In this section, I very briefly review a number of strategies that have been proposed in the AI literature for dealing with the difficulties associated with referring to the physical world. I then review a related research theme in language acquisition for an alternative perspective on this question. Finally I defend the position that communicating a task is about introducing the robot to a point of view, or perspective, from which that task is easy and well-organized, and the boundaries around objects and actions are more or less well delineated.

2.1 AI and the physical world

Since a robot's behavior is tightly linked to the world around it, the robot's internal representations will need to make reference to that world. For computational objects, as argued earlier, this is often a trivial matter. Perhaps objects can be described simply by an address. But what do you do if you live and work in a world where virtually nothing has a simple, clean, unambiguous description? Here are cartoon-caricatures of some of the answers AI researchers have developed:

- ▷ Avoid the need for referring to anything but raw sensor data. In simple cases, it may be possible to use the world as its own model (Brooks 1991b). Tasks such as obstacle avoidance can be achieved reactively, and Connell (1989) gives a good example of how a task with temporal structure can be performed by keeping state in the world and the robot's body rather than within its control system. This work clearly demonstrates that the structure of a task is distinct from the structures used to represent it. Activity that is organized around some external structure in the world does not imply a control system that directly references that structure.
- ▷ Adopt a point of view from which to describe the world that is sufficient for your task and which simplifies the kind of references that need to be made, hopefully to the point where they can be easily and accurately maintained. Good examples include deictic representations like those used in Pengi (Chapman & Agre 1987), or Toto's representations of space (Mataric 1990).
- ▷ Use multiple representations, and be flexible about switching between representations as each run into trouble (Minsky 1985). This idea overlaps with the notion of encoding common sense (Lenat 1995), and using multiple partial theories rather than searching – perhaps vainly – for single unified representations.

While there are some real conflicts in the various approaches that have been adopted, they also arguably have a common thread of pragmatism running through them. Some ask "what is the minimal representation possible", others "what choice of representation will allow me to develop my system most rapidly?" (Lenat 1995). For the tasking domain, a pragmatic question to ask is "what choice of representation will let the robot achieve its current task?"

The task-dependent "point of view" representations used by Pengi and Toto look attractive for our purposes. But if we wish to be able to request the robot to perform novel tasks, then it is not obvious how the robot can determine the appropriate point of view to adopt to describe that task appropriately. The key observation is that this is exactly the kind of information a human can provide and would expect to provide during tasking. We will view the process of requesting the robot to perform a novel task to fundamentally be all about communicating an appropriate point of view to it - how to segment the processes associated with the task, what to ignore and what to attend to.

At this point, it is useful to look at a parallel theme from the field of language acquisition.

2.2 The Poverty of the Stimulus

Problems of reference abound in philosophy and psychology, spawning whole fields of research such as semiotics. I will not even attempt to review this literature, but will instead trace out a single thread through it that has lead to results of some practical import for our purposes.

It has been observed that language acquisition involves a search through a large search space of models guided by relatively sparse feedback and few examples [need reference]. This so-called "poverty of the stimulus" relative to the complexity of the models being acquired is taken to imply that infants must have a good search strategy, with biases well matched to the nature of appropriate solution. This is a claim of innate constraints, and is historically controversial. I will not address that debate here since it seems entirely moot for our purposes.

Examples stressing under-determination in language learning include Quine's "Gavagai" example (Quine 1960). Quine invites us to consider walking with a native guide in a foreign country, and seeing a rabbit pass just as the guide says "gavagai". Without further evidence, Quine contends that the meaning of "gavagai" is underdetermined. It could mean "rabbit", "furry", "nice day, isn't it?", "undetached part of rabbit". This is an example of referential indeterminacy. He gave this example in the context of a "radical translator", who is working amongst people whose culture is entirely different to his own. But the basic problem is endemic, and also applies to infants who are just entering into the culture of those around them.

Another example is put forward by Goodman (1983) – he asks: why do we assume emeralds are green and not grue, where grue is "green when examined before the year 2001, and blue when examined thereafter"? Both are consistent with our observations (or were until the year 2001). This is the "new riddle of induction".

For our purposes, Quine's example highlights the importance of shared assumptions and protocol to at least narrow the class of possible interpretations of references. Goodman's example applies to all forms of induction. If we consider its implications for establishing shared references, we again see the importance of first having some shared assumptions and biases.

Tracing the strand on further, we come to theories such as that of Markman (1989) who propose particular constraints infants might use to map words on to meanings. These constraints are along the style of the following (with many variations, elaborations and caveats) :-

- Whole-object assumption. If an adult labels something, assume they are referring to the whole object and not a part of it.
- Taxonomic assumption. Organize meanings by "natural categories" as opposed to thematic relationships. For example when child is asked to find "dog", may fetch the cat, but won't fetch dog-food.
- Mutual exclusivity. Assume objects have only one label. So look for an unnamed object to apply a new label to.

These constraints are intended to explain the spurt in vocabulary acquisition, where infants acquire words from one or a few examples – so-called fast-mapping. They are advanced not as absolute rules, but as biases on search.

2.3 Shared perspective

Tomasello (1997) reviews some objections to the constraint-based approach represented by Markman. Tomasello favors a "social-pragmatic" model of language acquisition that places language in the context of other joint referential activity, just as shared attention. He rejects the "word to meaning mapping" formulation of

language acquisition. Rather, Tomasello proposes that language is used to invite others to experience the world in a particular way. From Tomasello (1997) :-

> The social-pragmatic approach to the problem of referential indeterminacy ... begins by rejecting truth conditional semantics in the form of the mapping metaphor (the child maps word onto world), adopting instead an experientialist and conceptualist view of language in which linguistic symbols are used by human beings to invite others to experience situations in particular ways. Thus, attempting to map word to world will not help in situations in which the very same piece of real estate may be called: "the shore" (by a sailor), "the coast" (by a hiker), "the ground" (by a skydiver), and "the beach" (by a sunbather).

Regardless of the utility of Tomasello's theory for its proper domain, language acquisition in infants, it seems a useful mindset for tackling robot tasking. When tasking, we will in essence be inviting the robot to view the world in a particular way, and hopefully one that is well suited to the task at hand.

Of more practical value for us, Tomasello and his collaborators developed a series of experiments designed to systematically undermine the constraints approach to learning as typified by Markman and others. The experiments investigate word learning among children in the context of various games. The experiments are instructive in showing a range of situations in which simple rules based directly on gaze or affect would fail in at least one case or other. They can therefore serve as prototypes for testing our progress. The experiments all avoid giving children (18-24 months old) ostentative naming contexts, and rather requiring them to pull out meanings from the "flow of interaction".

Scenario 1: Seeking with Success

In this experiment, an adult makes eye-contact with a child subject and says "Let's go find the toma." They then go to a row of buckets, each if which contains an object with which the child is not familiar. One of these objects is randomly designated the "toma". If the session is a control, the adult goes directly to the bucket containing the toma, finds it excitedly and hands it to the child. Otherwise, the adult first goes to two other buckets in sequence, each time taking out the object, scowling at it, and replacing it, before "finding" the toma. Later, the child is tested for for the ability to comprehend and produce the new word appropriately. The results show equally good performance in the test and control scenarios.

Tomasello argues that this situation counts against children using simple word learning rules such as "the object the adult is looking at while saying the novel word," "the first new object the adult looks at after saying the novel word," "the first new object the infant sees after hearing the novel word," or such variants. Regardless of the validity of the data for infant development, certainly if we want our robots to handle situations of this nature we must do better.

Scenario 2: Seeking with Disappointment

In this variation, the bucket containing the toma is replaced with a toy barn which could be closed. The adult and child first went around to the buckets and the barn, taking out objects and putting them back, the adult saying things like "let's see what's in here". After doing this for a while, the adult announces "now let's find the toma!" and proceeds directly to the barn. In the control condition, the adult takes out the object in the barn, and then moves on to the buckets. In the test condition, the adult attempts to open the barn but fails to do so, looking disappointed and saying "It's locked. I can't open it." Then as for the control, the adult moves on to the buckets to take out other objects.

Comprehension results again show equally good performance in the test and control condition. This case has the interesting property that the child never sees the "toma" after hearing its name. It has also been

replicated for 18 month olds. Tomasello argues that this result counts against word learning rules relying on excitement or positive affect. The adult exhibits only disappointment at the location of the target object, and excitement at the adjacent bucket – but this misleads very few children.

Scenario 3: Novelty

In another experiment, the child, its parent, and two experimenters played together with three novel, unnamed objects. After some time, the parent and one of the experimenters left the room. The remaining experimenter then introduced a new object to the child and they played with it for the same length of time as for the other objects. Then the experimenter placed the objects in a row (optionally after playing with some other toy rather than the novel one for a period), at which point the others returned. In the test condition, the adults looked towards the objects without singling one out and said excitedly "Look, I see a gazzer! a gazzer!" In the control condition, they instead said "Look, I see a toy! A toy!".

Comprehension tests showed that children could learn the word correctly in this condition, assigning the "gazzer" label to the toy that was novel to the returning adults.

2.4 Summary

So what can we conclude? If we want robots to be able to cope with scenarios like these, they will need a deep understanding of the activities around them. We can treat the range of naming situations a robot can deal with as a test of the depth of that understanding. For example, consider search. If the robot understands the purpose of searches, how they succeed and fail, then that will naturally extend the range of naming situations it can deal with beyond simple ostentive associations. In the infant development literature, considerable emphasis is placed on the child's ability to interpret the behavior of others in terms of intent using a "theory of mind". Such an ability is very powerful, but difficult to implement. As a precursor, I consider an initial interpretation of activity in terms of process (branches, loops, sequences) rather than intent. This is sufficient to establish an initial shared perspective between human and robot.

3 Communicating task structure

While individual parts of a task may be difficult to describe formally, its abstract structure or control flow will often be amenable to description. For example, the overall branch-and-loop flow of a sorting task is easily expressed, but the actually sorting criterion may depend on differentiating two classes of objects based on a small difference in their appearance that would be easier to demonstrate than to describe. If we go ahead and communicate the task structure to the robot, it can be used to guide interpretation of the less easily expressed components. Figure 1 shows a schematic for how this may be achieved. The basic idea is for the robot to interact with the instructor vocally to acquire a "sequencing model" of that task, and then to ground that model based on a demonstration of the task. The demonstration is annotated by the instructor both in terms of the sequencing model and in terms of previously grounded elements.

In this section, I concentrate on the "task modeling" module in figure 1. This is a mechanism for communicating task structure to the robot. This is composed of two parts :-

1. A "sequencing protocol" describing how the human tutor needs to behave to communicate the task structure, and the feedback the tutor can expect from the robot.
2. A sequencing module that supports the robot's side of this protocol, correctly recovering the task structure from the tutor's behavior and providing the appropriate feedback.

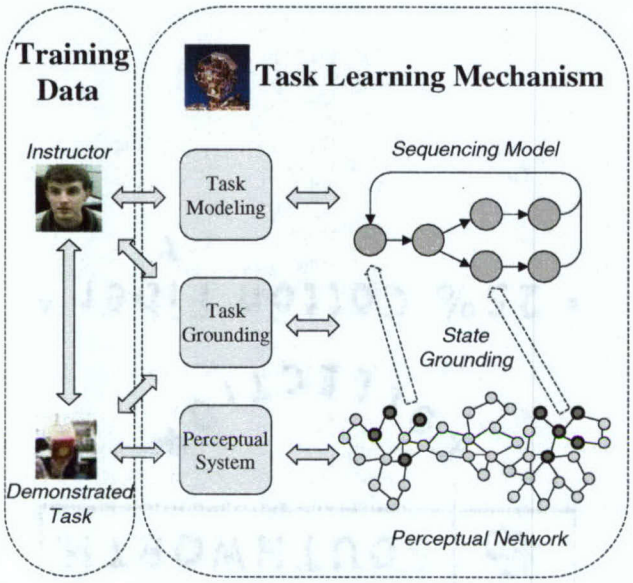

Figure 1: Tasking summary.

3.1 The sequencing protocol

Ideally, the sequencing protocol would simply require natural human behavior. Borchardt (1993) has demonstrated that models of physical activities can be reconstructed from descriptions that follow Grice's conversational maxims. Making use of this for a physical robot would require a long excursion into research far removed from the thrust of this thesis. Previous and ongoing work at the Humanoid Robotics Group at the AI Lab has addressed the question of whether robots can evoke particular speech registers and behavior patterns that simplify perception (Breazeal 2000, Varchavskaia & Fitzpatrick 2001). For the purposes of this thesis, I will be happy with a protocol that can be supported by a human with a small amount of prior instruction. This does not mean the protocol can be arbitrarily alien. For example, the author's prior experience with the robot Kismet suggests that it is almost impossible for most people to remember that the robot can only hear them if they speak into a wireless microphone, no matter how often they are reminded of its (unfortunate) necessity. The protocol will have to lie along the grain of natural human behavior.

Currently, I have implemented the sequencing protocol through verbal annotation. As the human tutor demonstrates a task, they are expected to verbalize their activity. Initially the robot cannot make much of the demonstration, but it can process the speech stream, and attempt to recover the structure of the task from that. In particular, the robot will attempt to determine "states" of the task – points at which the demonstration returns to what is effectively the same mode. How it does that will be addressed in the next section, but clearly the more the annotations actually correspond to states the easier this will be. In my current implementation, the annotations are simple vocal labels corresponding to actions, configurations, objects, or whatever the tutor finds mnemonic. Section 6 will discuss a more flexible way to deal with speech.

Briefly, the most practical method to deal with speech is to develop a "command-and-control" vocabulary, where what the human may utter is highly constrained and so fewer competing hypotheses need be entertained during recognition. For a deployed, practical system this remains about the only workable solution. Zue & Glass (2000) gives a good overview of the state of the art in speech processing. For this work, I am moving beyond this in two ways. First, convenient "throw-away" vocabulary can be introduced by the tutor to match the idiosyncratic objects and actions used within a task. Second, "filler" grammar can be detected and modeled as will be described in section 6.

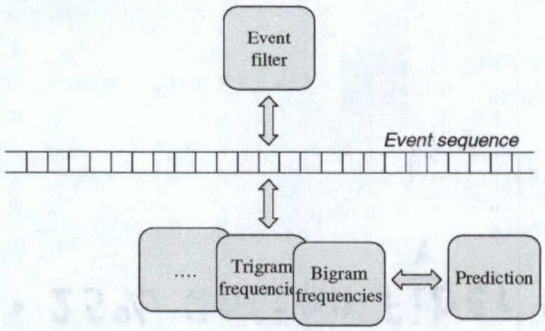

Figure 2: Using n-grams for task modeling.

For the remainder of the discussion, I will assume that speech can be converted to a sequence of discrete events that may or may not be state-like. This is so that if other demonstrative acts can be detected reliably, they can be used interchangeably with the vocalizations for the purposes of annotation.

3.2 The sequencing module

The key aspect of recovering the task model is that it must be easy. If the task model is hard to recover, then it is purposeless – one might as well bypass it altogether. If the human side of the sequencing protocol has been correctly designed, we should not need to invoke sophisticated learning mechanisms.

The method I have used so far draws on n-gram modeling procedures developed for speech recognition [ref]. Here, we estimate the probability of event sequences from models trained on sequence frequency counts from a corpus. Models vary in the amount of history they incorporate – bigram models, trigram models etc. Low order models are limited in the dependencies they can capture, but can be trained up with relatively little data. High order models are more expressive but harder to train. Best results are achieved when n-gram models of many orders are used, and interpolated based on the amount of training data available for each context. A criticism of such models is that they cannot capture long-distance dependencies, but it has proven difficult to do much better for the purposes of recognition (generation is another matter).

Figure 2 shows the system organization. An event filter pulls out candidate states from the instructor's vocalizations. A collection of n-gram models are constructed online as the events arrive. Predictions are continuously made for the next event using an interpolation scheme, as discussed above. One addition is that incorrect predictions are fed back to the n-gram models, so that statistics can be maintained both for sequences that occur and also for sequences that do not occur but which the system might otherwise predict from lower order statistics. This does not change the order of the number of sequences for which statistics are kept, and greatly improves the performance of the algorithm when data is scarce.

Not shown in the system diagram are mechanisms to deal with back-tracking and forgetting. Updates to the model are made in a way that can be rolled back if the user signals an error in the robot's understanding. Multiple copies of all the models are constructed, each beginning at small offsets in time. By switching between these models in a round robin fashion, we can reset the models periodically to effectively give the robot a finite working memory without introducing glitches in performance or requiring pauses for re-training.

I plan to also investigate using augmented markov models for task modeling (Goldberg & Mataric 1999). But by design, there is no need for a complicated learning mechanism here.

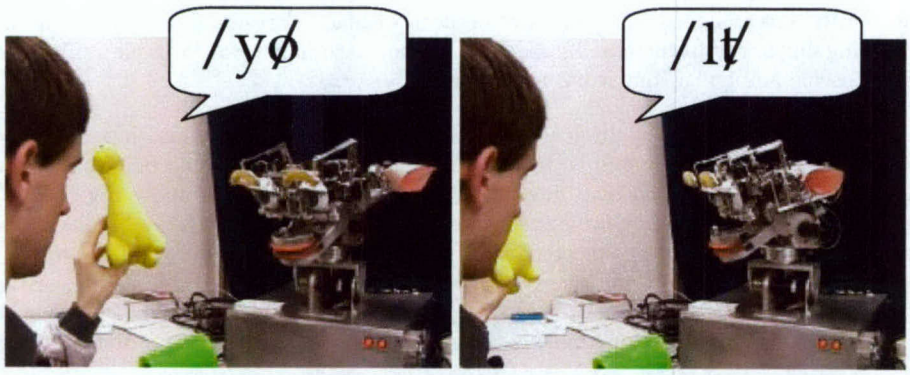

Figure 3: Two situations, distinguished principally by process state. On the left, an object is presented to the robot and its color is described vocally. Then the object is moved in one direction or another based on its color, and the direction is described. The number of possible interpretations of what the vocalization might refer to is vast, despite at least being filtered by recursive attention. But once the "shape" of the process is available, the ambiguity is reduced.

4 Grounding the task

Once the robot has the beginnings of a model for the task structure, as discussed in the previous section, it can begin to relate that to the actual physical demonstration the human tutor is making. Machine perception is necessarily noisy and full of ambiguity. The degree to which this is so for a given task will fundamentally limit the complexity of any modeling the robot can do, if we permit uncertainty to compound on uncertainty. By first establishing a task model through a relatively noise-free protocol for which we can depend on error-correcting feedback from the human tutor, we limit the impact that uncertainty in grounding one element of the model will have on all the others – at least while the task is being introduced, though not of course when the robot is performing it.

The task structure recovered in the previous section allows us to pose grounding as solving a set of supervised learning tasks, with examples identified by annotations, and the number of classes by the task's branching structure. An example is shown in figure 3. To actually solve such learning tasks, I will use a basic, well-understood function approximator such as support vector machines. The thrust of this thesis is not to develop the best approximator and training procedure possible, but rather to make it work in concert with the task structure recovered in the previous section, and (crucially) the feature selection process to come in the next section. The goal is again to make this machine learning step as trivial as possible – to make it essentially "communicative" in nature rather than what is normally thought of as "learning". In general my strategy is that if the robot is ever facing a hard learning problem, then there needs to be more communication. It is of course desirable for robots to be able to autonomously solve complex learning tasks, but for rapid tasking this should really be a last resort.

5 Communicating feature choice

At this point, we have successfully communicated the overall structure of the task to the robot, so that it knows when the decision points within the task occur. If those decisions are based on signals strongly and consistently represented in the robot's perceptual network, the grounding process will eventually determine this. In this section, I look at how we can speed up this process by essentially performing feature selection for the supervised learning mechanism.

What can we identify unambiguously for the robot in physical space? Presumably we can point out a specific location, by moving our fingertip there and making an appropriate signal (for example). There are problems with this, such as scale ambiguity, but it has some value.

Can we refer to objects? The idea of a physical object is rarely completely coherent, since it depends on where you draw its boundary and that may well be task-dependent. One solution would be to use a convention; we might agree that the object indicated by a fingertip is that mass which would move together if you pushed at that point. This definition has the attractive property of breaking down into ambiguity in close to right circumstances, such as for large interconnected constructs, floppy formless ones, liquids etc. Of course, the robot needs a way to make the determination of what will move together – this is addressed in section 7. And there are some difficulties with respect to separate objects that happen to be in contact in the direction of the push, so the definition needs to be make a little smarter, but it will do for a first approximation.

Can we refer to parts of objects? We can certainly refer to locations on the surface of the object, once we can refer to the object itself to rule it out as the referent. Again a scale or boundary ambiguity remains, but may be bearable depending on the task.

Once we establish a reference to an object or location, we can maintain it through a level of indirection by binding it to some communicative gesture or utterance, or by coarse location references (such as a vague glance or pointing gesture) that rely on the robot being able to complete the reference from memory. We will use utterances as our primary indirect reference mechanism, but will also make use of coarse location references as confirming gestures used by both the human and robot.

Can we refer to physical properties of objects, such as color, shape, texture, weight, sturdiness? We could imagine developing conventions for each type of property, just like we have a special convention for referring to location. Binding such properties to utterances seems the most practical solution. The bindings do not need to be hardwired; they can be communicated through a task where the desired property is critical. The slow grounding process will eventually pick out the property after many examples, assuming it is expressed in a direct and consistent way in the robot's perceptual network, and at that point we can bind a vocalization to it. From then on, we can select out that feature vocally rather than having to give many examples.

But there is a faster way to refer to features. If we assume the robot characterizes the statistical properties of the elements of its perceptual network, then we can "point" into that network by violating those statistics. We can present extreme values of a property, such as a bright saturated green. We can present repeated extreme contrasts of a property, such as distance. We can synchronize such presentations with vocalizations, to make their temporal correlation with that special signal stand out. All of these are useful and important conditions for the robot to be checking for in any case, and there is nothing to stop us using them with an explicitly communicative intent.

Can we refer to actions? We can certainly identify the time that they occur at, and the objects and properties involved. Every action is a miniature task, so referring to one may require demonstrating it on its own, broken down into further states until grounding is possible. This brings up the problem of incorporating sub-tasks in larger tasks. Referring to a task is more than an issue of binding it to a vocalization, since to be useful it probably needs to be parameterized. Joint attention mechanisms are other implicit state can remove some parameters but is not sufficient in general. So there needs to be an equivalent of "procedure calls" with arguments in the sequencing protocol.

We may also consider the human making reference to an action of the robot, or to an object the robot is manipulating. Here binding is potentially simpler. This is analogous to early infant-caregiver interaction, where the caregiver interprets and responds to the behavior of the infant, gradually leading it outwards to interpreting and responding to others.

All the above procedures are intended to operate with minimal initial shared vocabulary. This isn't necessary, but is good for exposing inflexibility in the mechanisms.

Use of vocabulary

Throughout the above, we have mentioned introducing a more direct means of indexing, in the form of a shared vocabulary. In fact this mechanism provides quite a robust way to ground labels simply as sensitivities to features. The meaning of labels of this type can be referred back directly to the robot's feature vector, giving natural semantics for combination, contrast, etc.

Such meanings are not very interesting yet. Nevertheless, we have some power. A tutor could walk up to a robot, and show it a contrasting condition such as "static, moving, static, moving" a few times. Assuming the robot represents gross motion in one or a few features, this would be picked out (along with any correlated features). The tutor can associate the selected features with a label, and later reevoke them (or less importantly, reevoke the particular *values* associated with "static" or "moving" – less importantly because particular values can easily be recovered by supervised learning once the appropriate features are known). The labels themselves can be treated as corresponding to a virtual property.

Things become interesting when we consider what happens if the tutor tries to communicate a property that the robot does not represent in a simple manner. In this case, no single part of the robot's perceptual network will be strongly activated by the tutor's behavior. The only thing to do is fall back on supervised learning across the full perceptual network, using the tutor's behavior to provide labels for the network's state at various instants. A robot with high bandwidth sensors will presumably have a large network so this is apt to be slow. If the tutor realizes this, then they can still help by aiding in feature selection. Any features or groups of features that have been previously established can be reevoked, and their union used to bias weights on the features during the learning process. We don't have to specify *how* the features might interact – conjunction, mutual exclusion, absence, or any of the many possibilities. The function approximator is left to deal with that, and is likely to succeed since we have done the hard part of feature selection. Hence the labels have robust grounding – their combined semantics is free to adapt to each particular usage up to the expressive power of the search space of the function approximator.

6 Speech Protocol

For speech processing, I intend to use a simple "command and control" style interface using the MIT SLS group's SUMMIT speech recognizer (Glass et al. 1996), augmented to deal with a growing vocabulary. I describe a technique to bootstrap from an initial vocabulary by building an explicit model of unrecognized parts of utterances. The purpose of this background model is both to improve recognition accuracy on the initial vocabulary and to automatically identify candidates for vocabulary extension. This work draws on research in word spotting and speech recognition. I bootstrap from a minimal background model, similar to that used in word-spotting, to a much stronger model where many more word or phrase clusters have been "moved to the foreground" and explicitly modeled. This is intended both to boost performance on the original vocabulary by increasing the effectiveness of the language model, and to identify candidates for automatic vocabulary extension.

The remainder of this section shows how a conventional speech recognizer can be convinced to cluster frequently occurring acoustic patterns, without requiring the existence of transcribed speech data.

Clustering algorithm

A speech recognizer with a phone-based "OOV" (out-of-vocabulary) model is able to recover an approximate phonetic representation for words or word sequences that are not in its vocabulary. If commonly occurring phone sequences can be located, then adding them to the vocabulary will allow the language model to capture their co-occurrence with words in the original vocabulary, potentially boosting recognition performance.

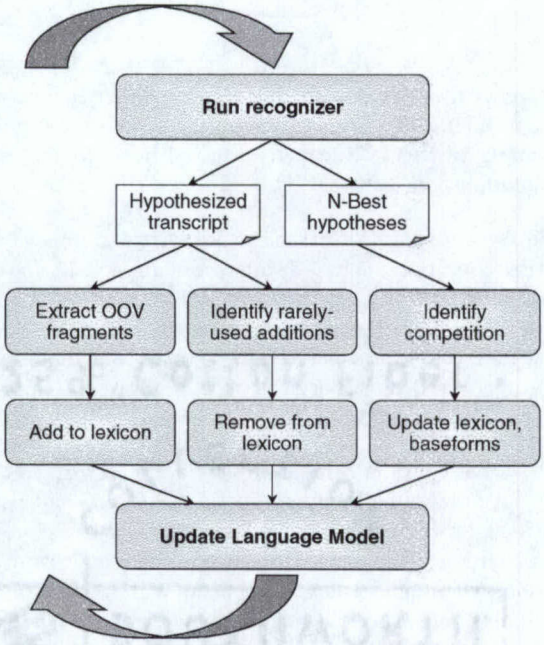

Figure 4: The iterative clustering procedure.

This suggests building a "clustering engine" that scans the output of the speech recognizer, correlates OOV phonetic sequences across all the utterances, and updates the vocabulary with any frequent, robust phone sequences it finds. While this is feasible, the kind of judgments the clustering engine needs to make about acoustic similarity and alignment are exactly those at which the speech recognizer is most adept.

The clustering procedure adopted is shown in Figure 4. An *n*gram-based language model is initialized uniformly. Unrecognized words are explicitly represented using a phone-based OOV model, described in the next section. The recognizer is then run on a large set of untranscribed data. The phonetic and word level outputs of the recognizer are compared so that occurrences of OOV fragments can be assigned a phonetic transcription. A randomly cropped subset of these are tentatively entered into the vocabulary, without any attempt yet to evaluate their significance (e.g. whether they occur frequently, whether they are similar to existing vocabulary, etc.). The hypotheses made by the recognizer are used to retrain the language model, making sure to give the new additions some probability in the model. Then the recognizer runs using the new language model and the process iterates. The recognizer's output can be used to evaluate the worth of the new "vocabulary" entries. The following sections detail how to eliminate vocabulary items the recognizer finds little use for, and how to detect and resolve competition between similar items.

Extracting OOV phone sequences

I use the speech recognizer system developed by the SLS group at MIT (Glass et al. 1996). The recognizer is augmented with the OOV model developed by Bazzi in (Bazzi & Glass 2000). This model can match an arbitrary sequence of phones, and has a phone bigram to capture phonotactic constraints. The OOV model is placed in parallel with the models for the words in the vocabulary. A cost parameter can control how much the OOV model is used at the expense of the in-vocabulary models. This value was fixed at zero throughout the experiments described in this paper, since it was more convenient to control usage at the level of the language model. The bigram used in this project is exactly the one used in (Bazzi & Glass 2000), with no training for the particular domain.

Phone sequences are translated to phonemes, then inserted as new entries in the recognizer's lexicon.

Dealing with rarely-used additions

If a phoneme sequence introduced into the vocabulary is actually a common sound sequence in the acoustic data, then the recognizer will pick it up and use it in the next iteration. Otherwise, it just will not appear very often in hypotheses. After each iteration a histogram of phoneme sequence occurrences in the output of the recognizer is generated, and those below a threshold are cut.

Dealing with competing additions

Very often, two or more very similar phoneme sequences will be added to the vocabulary. If the sounds they represent are in fact commonly occurring, both are likely to prosper and be used more or less interchangeably by the recognizer. This is unfortunate for language modeling purposes, since their statistics will not be pooled and so will be less robust. Happily, the output of the recognizer makes such situations very easy to detect. In particular, this kind of confusion can be uncovered through analysis of the N-best utterance hypotheses.

If we imaging aligning a set of N-best hypothesis sentences for a particular utterance, then competition is indicated if two vocabulary items exhibit both of these properties:

- ▷ Horizontally repulsive - if one of the items appears in a single hypothesis, the other will not appear in a nearby location within the same hypothesis
- ▷ Vertically attractive - the items frequently occur in the same location within different hypotheses

Since the utterances in this domain are generally short and simple, it did not prove necessary to rigorously align the hypotheses. Instead, items were considered to be aligned based simply on the vocabulary items preceding and succeeding them. It is important to measure both the attractive and repulsive conditions to distinguish competition from vocabulary items that are simply very likely to occur in close proximity.

Accumulating statistics about the above two properties across all utterances gives a reliable measure of whether two vocabulary items are essentially acoustically equivalent to the recognizer. If they are, they can be merged or pruned so that the statistics maintained by the language model will be well trained. For clear-cut cases, the competing items are merged as alternatives in the list of pronunciation variants for a single vocabulary unit. or one item is simply deleted, as appropriate.

Here is an example of this process in operation. In this example, "phone" is a keyword present in the initial vocabulary. These are the 10-best hypotheses for the given utterance:

"what is the phone number for victor zue"

```
<oov> phone (nahmber) (mihterz) (yuw)
<oov> phone (nahmber) (mihterz) (zyuw)
<oov> phone (nahmber) (mihterz) (uw)
<oov> phone (nahmber) (mihterz) (zuw)
<oov> phone (ahmberf) (mihterz) (zyuw)
<oov> phone (ahmberf) (mihterz) (yuw)
<oov> (axfaanah) (mberfaxr) (mihterz) (zyuw)
<oov> (axfaanah) (mberfaxr) (mihterz) (yuw)
<oov> phone (ahmberf) (mihterz) (zuw)
<oov> phone (ahmberf) (mihterz) (uw)
```

The "<oov>" symbol corresponds to an out of vocabulary sequence. The sequences within parentheses are uses of items added to the vocabulary in a prior iteration of the algorithm. From this single utterance, we acquire evidence that:

- ▷ The entry for (ax f aa n ah) may be competing with the keyword "phone". If this holds up statistically across all the utterances, the entry will be destroyed.
- ▷ (n ah m b er), (m b er f axr) and (ah m b er f) may be competing. They are compared against each other because all of them are followed by the same sequence (m ih t er z) and many of them are preceded by the same word "phone".
- ▷ (y uw), (z y uw), and (uw) may be competing

All of these will be patched up for the next iteration. This use of the N-best utterance hypotheses is reminiscent of their application to computing a measure of recognition confidence in (Hazen & Bazzi 2001).

Testing for convergence

For any iterative procedure, it is important to know when to stop. If we have a collection of transcribed utterances, we can track the keyword error rate on that data and halt when the increment in performance is sufficiently small. Keywords here refer to the initial vocabulary.

If there is no transcribed data, then we cannot directly measure the error rate. We can however bound the rate at which it is changing by comparing keyword locations in the output of the recognizer between iterations. If few keywords are shifting location, then the error rate cannot be changing above a certain bound. We can therefore place a convergence criterion on this bound rather than on the actual keyword error rate. It is important to just measure changes in keyword locations, and not changes in vocabulary items added by clustering.

6.1 Preliminary explorations in vocabulary extension

The unsupervised procedure described in the previous section is intended to both improve recognition accuracy on the initial vocabulary, and to identify candidates for vocabulary extension. This section describes experiments that demonstrate to what degree these goals were achieved. To facilitate comparison of this component with other ASR systems, results are quoted for a domain called LCSInfo (Glass & Weinstein 2001) developed by the SLS group at MIT. This domain consists of queries about personnel – their addresses, phone numbers etc. Very preliminary results for Kismet-directed speech are also given.

Experiment 1: qualitative results

This section describes the candidate vocabulary discovered by the clustering procedure. Numerical, performance-related results are reported in the next section.

Results given here are from a clustering session with an initial vocabulary of five keywords (email, phone, room, office, address), run on a set of 1566 utterances. Transcriptions for the utterances were available for testing but were not used by the clustering procedure. Here are the top 10 clusters discovered on a very typical run, ranked by decreasing frequency of occurrence:

1	n ah m b er	6	p l iy z
2	w eh r ih z	7	ae ng k y uw
3	w ah t ih z	8	n ow
4	t eh l m iy	9	hh aw ax b aw
5	k ix n y uw	10	g r uw p

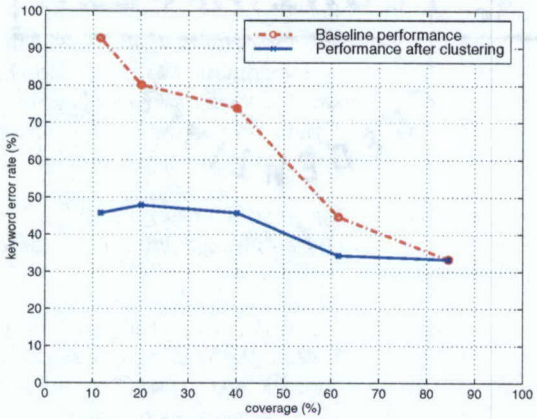

Figure 5: Keyword error rate of baseline recognizer and clustering recognizer as total coverage varies.

These clusters are used consistently by the recognizer in places corresponding to: "number, where_is, what_is, tell_me, can_you, please, thank_you, no, how_about, group," respectively in the transcription. The first, `/n ah m b er/`, is very frequent because of phrases like "phone number", "room number", and "office number". Once it appears as a cluster the language model is immediately able to improve recognition performance on those keywords.

Every now and then during clustering a "parasite" appears such as `/dh ax f ow n/` (from an instance of "the phone" that the recognizer fails to spot) or `/iy n eh l/` (from "email"). These have the potential to interfere with the detection of the keywords they resemble acoustically. But as soon as they have any success, they are detected and eliminated as described earlier. It is possible that if a parasite doesn't get greedy, and for example limits itself to one person's pronunciation of a keyword, that it will not be detected, although we didn't see any examples of this happening.

Experiment 2: quantitative results

For experiments involving small vocabularies, it is appropriate to measure performance in terms of Keyword Error Rate (KER). Here this is taken to be:

$$KER = \frac{F + M}{T} * 100 \tag{1}$$

with:

F	=	Number of false or poorly localized detections
M	=	Number of missed detections
T	=	True number of keyword occurrences in data

A detection is only counted as such if it occurs at the right time. Specifically, the midpoint of the hypothesized time interval must lie within the true time interval the keyword occupies. We take forced alignments of the test set as ground truth. This means that for testing it is better to omit utterances with artifacts and words outside the full vocabulary, so that the forced alignment is likely to be sufficiently precise.

The experiments here are designed to identify when clustering leads to reduced error rates on a keyword vocabulary. Since the form of clustering addressed in this paper is fundamentally about extending the

vocabulary, we would expect it to have little effect if the vocabulary is already large enough to give good coverage. We would expect it to offer the greatest improvement when the vocabulary is smallest. To measure the effect of coverage, a complete vocabulary for this domain was used, and then made smaller and smaller by incrementally removing the most infrequent words. A set of keywords were chosen and kept constant and in the vocabulary across all the experiments so the results would not be confounded by properties of the keywords themselves. The same set of keywords were used as in the previous section.

Clustering is again performed without making any use of transcripts. To truly eliminate any dependence on the transcripts, an acoustic model trained only on a different dataset was used. This reduced performance but made it easier to interpret the results.

Figure 5 shows a plot of error rates on the test data as the size of the vocabulary is varied to provide different degrees of coverage. The most striking result is that the clustering mechanism reduces the sensitivity of performance to drops in coverage. In this scenario, the error rate achieved with the full vocabulary (which gives 84.5% coverage on the training data) is 33.3%. When the coverage is low, the clustered solution error rate remains under 50% - in relative terms, the error increases by at most a half of its best value. Straight application of a language model gives error rates that more than double or treble the error rate.

As a reference point, the keyword error rate using a language model trained with the full vocabulary on the full set of transcriptions with an acoustic model trained on all available data gives an 8.3% KER.

Experiment 3: Kismet-directed speech

An experiment was carried out for data drawn from robot-directed speech collected for the Kismet robot. This data comes from an earlier series of recording sessions for the work described in (Breazeal & Aryananda 2000). Early results are promising – semantically salient words such as "kismet", "no", "sorry", "robot", "okay" appear among the top ten clusters. But this work is in a very preliminary stage, since an acoustic model needs to be trained up for the robot's microphone configuration and environment.

7 Active segmentation

In section 2, physical objects were defined operationally as whatever moves together when prodded. This is not something that can be recovered easily from vision, since the problem of object segmentation is a difficult one which in general needs to be solved in conjunction with object recognition. An alternative is to use distance information, for example from stereo vision. On the timeframe of this thesis, I expect to have a multiple-baseline robot head available to work with. But the unique advantage robots have for object segmentation is that they can reach out and touch the world. Imagine the classical face/vase illusion – this is trivial to resolve if you can simply poke it to see which part is free space and which is not. In this section I enumerate active strategies a robot can use for achieving object segmentation. I wish to implement these, as prototypes of the kind of physical querying that is one of the "killer apps" of robotics. For simplicity of implementation, the strategies are constrained to rely on a robot arm only – no hand or fingers.

If the robot is unsure where the boundaries of an object lie, here are some strategies it can use :-

1. Poke the object gently. Tapping a solid object will induce a small motion of that object. This will result in a coherent region of optic flow on the image plane. If the object is non-rigid, or attached to other objects, then the response will be messy and complicated – but this is in some sense inevitable, since it is in just such cases that the idea of a unique "object boundary" runs into trouble
2. Thump the object savagely. A big disturbance is apt to generate a confusing motion that is hard to process directly. But it will move the object away from its local surroundings, giving another "role of

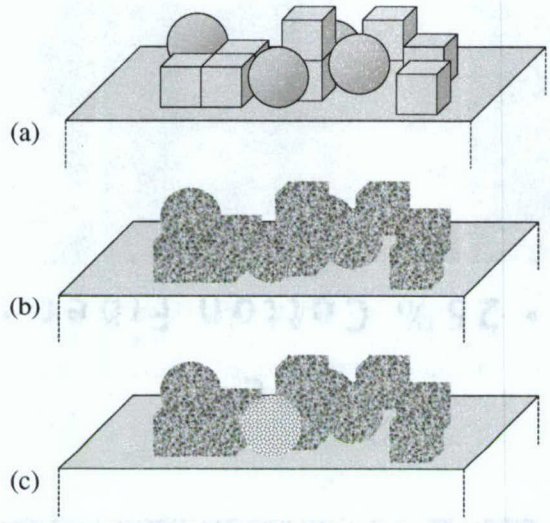

Figure 6: A cartoon suggesting how clear coherent motion can be to the robot. (a) shows a collection of objects as they appear to a human. (b) shows how they might appear to a robot if they do not have strong variation in color or texture. (c) shows how an object "lights up" a motion filter when tapped.

the dice" – an opportunity for the robot to see the object against a new background, perhaps with better contrast. Frequently visual ambiguity is only a local, accidental effect.

3. Try to get the arm's endpoint beside the object. Anywhere the endpoint can reach is presumably free space, constraining the boundary of the object. We can use the arm's endpoint as a mobile reference object to confirm our theories of where free space lies.

4. Try to get the arm's endpoint behind the object. This has the advantage of putting a known background behind the object. Imagine the arm painted bright red to see the advantage of this for identifying the object's boundary.

5. Ask the human to present the object. A human bringing an object near the robot offers the dual advantage of motion cues and a known (if complicated) partial background – the hand.

6. One remaining alternative is to displace the robot's own head and body, again to get another "role of the dice", or to access three-dimensional information over a longer baseline than is available from the stereo cameras.

8 Gaze protocol

Gaze is important for spatial localization. It is crucial that the robot carefully controls its own gaze to convey the most veridical impression of its state to the human. It is useful if the robot is also sensitive to the gaze of the human since it is informative of the human's locus of attention. We address both directions here.

Figure 7: These images are from a sequence in which the instructor wanted the robot to attend to the green object as it moved away from a central location. In the first image the robot is clearly attending; in the second it just as clearly has become fixated on the instructors face. Knowing this prompted the instructor to wave the object a little until it regained the robot's attention.

8.1 The robot's gaze

It helps if the robot moves its eyes in a manner consistent with human eye movement. I did work on this in collaboration with Brian Scassellati and Cynthia Breazeal (Breazeal, Edsinger, Fitzpatrick, Scassellati & Varchavskaia 2000). Eye movement is of primary importance because it creates a very visible failure mode.

Extensions made beyond the work cited for this thesis will include a short-term memory for locations, so robot can look back at objects, or refer to objects or actions by looking towards a place it remembers it being associated with.

8.2 The human's gaze

For tasking, it is useful to track the gaze of the human tutor. In particular it is important to know whether the tutor is facing the robot, looking away, or looking at something near the robot. For robot tasking, I assume that periods of face-to-face contact occur, punctuated by whatever arbitrary movements the instructor needs to make.

First, I develop a 6-dimensional coordinate system that is convenient for tracking, deferring until later how those coordinates relate to the rigid body parameters we really want. The goal was to develop a coordinate system that isolates the impact of estimates of the shape of the head as much as possible. In fact, the tracking scheme here will work on concave objects of all sorts. The kinds of asymmetric, extended, awkwardly-shaped objects it would not work on are exactly the kind of objects for which pose can be determined relatively unambiguously from static images.

If an object being viewed does not rotate in depth but is otherwise free to explore a 3D space, there are four numbers that both describe the object's pose completely and are relatively easy to recover from the perspective projection of the object on the image plane. These are :-

- A position on the image plane, specified by two coordinates, giving a ray from the camera to the object.
- A coordinate specifying any in-plane rotation of the object, which is the only rotational freedom it has.
- A coordinate specifying a scaling of the projection of the object on the image plane.

These coordinates completely describe the object's pose in the sense that if the camera configuration is known, and *if the shape of the object is known*, the full 3D pose of the object can be recovered from these parameters. The need for the shape of the object arises from the implicit reference points of the coordinates

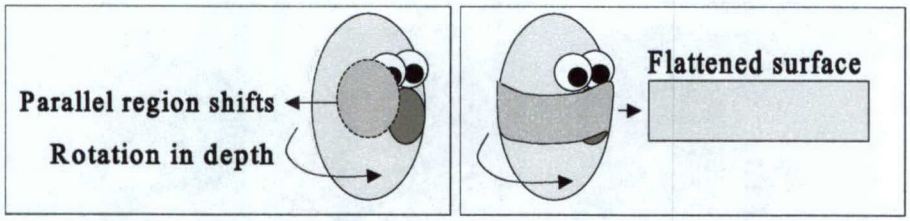

Figure 8: Left: when the head rotates in depth, a different region on the head will become parallel to the image plane. Right: if the regions of the head that become parallel during a movement of the head do not explore the surface in 2D, then the surface they do explore can be thought of as Euclidean without running into contradictions (except for full 360° excursions).

– for example, the scaling coordinate can be converted to distance only with knowledge of the size of the object.

Once the object starts rotating in depth, there are two more degrees of freedom to factor in. The goal here to introduce them without destroying the simplicity of the image plane coordinates defined above. Importing some domain knowledge, assume the object being tracked is basically convex. Then at any moment there will be a unique region on the surface of the object that is close to parallel to the image plane. As the object rotates in depth, this region will shift to another part of the surface. We can parameterize where this region lies on the surface of the object using two dimensions. And since the region is (by construction) parallel to the image plane, the four coordinates developed earlier can be recast as follows :-

- ▷ Two coordinates that specify where the projection of the parallel region lies on the image plane.
- ▷ A coordinate specifying how the parallel region is rotated with respect to the image plane. This is the only rotational degree of freedom the parallel region has, by construction.
- ▷ A coordinate specifying a scaling of the parallel region (or equivalently of the projection of the entire object, as before).

Combined with two coordinates that determine what part of the surface of the object is currently parallel to the image plane, we have a 6-dimensional coordinate system that fully specifies the 3D pose of the object (if the shape of the object is known). This choice of coordinates has some virtues. In contrast to Euler angles, for example, the coordinates can be considered separately and in any order. This is least obvious for the rotation coordinate, but becomes clear if that coordinate is thought of as a counter-rotation of the camera about its optical axis.

A crucial issue that has not yet been addressed is what kind of coordinates are used to span the surface of the object being tracked. There are many possible coordinate systems for specifying a location on a convex surface – for example, latitude and longitude angles. The challenge here is to use coordinates that can be related to the projection of the object without knowledge of its 3D shape. There is no such magical coordinate system, so technically at this point the dimensions of the head have to be estimated before proceeding any further. But suspending disbelief for a moment, consider setting up a Euclidean coordinate system on the surface (which can be thought of as flattening the surface out onto a plane and then using standard rectangular coordinates). Of course, it isn't possible to flatten out the surface in this way without introducing inconsistencies. But if we do so anyway, then coordinates on the surface of the object that lie within the parallel region will map on to the image plane very simply, with just a scaling and an in-plane rotation. If we only ever try to relate coordinates within this region, then we can relate small steps in the image plane to small steps on the surface of the object, and so integrate the surface coordinates without needing to know the actual shape of the object.

The above discussion imposes two conditions :-

Figure 9: Mesh initialization. The mesh is initially distributed arbitrarily. It is pruned by the head outline when it is detected, and by heuristics based on the relative motion of different parts of the mesh. When frontal pose is detected, surface coordinates of the mesh can be initialized within the parallel region (that part of the face that is parallel to the image plane).

1. We must be able to determine what part of the projection of the object originated from a surface parallel to the image plane.
2. The path the parallel region traces across the surface of the object must lie within a strip that is thin relative to the curvature of the object. The wider the strip, the less Euclidean it is. The strip also must not make a full 360° excursion, no matter how thin it is.

The first condition is tractable and will be addressed shortly. With regard to the second condition: in practice, the estimated curvature of the object should be factored in to the surface coordinate system, and this becomes an argument about what kinds of movements the accuracy of the estimate actually matters for. The answer is as might be expected: tracking accuracy is insensitive to the estimate of shape for movements combining in-plane translation, scaling (translation in depth), in-plane rotation, and rotation in depth for which all the surface patches made successively parallel to the image plane lie within a strip. This includes the important case of turning away, then turning back in an approximately symmetric manner.

To actually implement a pose tracking system based on this coordinate system, a mesh is laid down on the projecting of the head as illustrated in Figure 9. Nodes on the mesh are kept in correspondence to the face using simple template trackers, which are destroyed if they misbehave (measured by a set of consistency checks) and recreated elsewhere. Scaling, in-plane rotation, and in-plane translation are straightforward to compute from deformations of this mesh. As the head rotates in depth, some trackers will lose the support of the surface they are tracking as it becomes occluded, and so be destroyed. New parts of the head will become visible and have trackers assigned to their surface.

The mesh is used to maintain the surface coordinate system as follows. First, the parallel region is determined heuristically. If the translational component of motion can be eliminated, the parallel region can be identified easily because the flow due to rotation peaks there (since the motion of that surface is completely parallel parallel to the image plane). Translational motion can be accounted for by normalizing flow relative to the outline of the head. This crude procedure works better than it should because in practice translations and rotations of the head are often coupled so as to sum within the parallel region rather than cancel. Exceptions include pure rolls and translations in depth. The extent of the parallel region is chosen to scale in a heuristic way with the head outline, since in theory it should be infinitesimally small but in practice it has to be assigned some extent to be useful. And luckily, surface distortions such as the nose don't seem to cause trouble.

The parallel region can be seen as a mask overlaid on the image, within which it is safe to relate image coordinates and surface coordinates. Pose recognition events, detected in the manner described in the previous section, are used to choose an origin on the surface, and an initial translation, scaling and (in-plane) rotation of the surface coordinate system with respect to the image plane. This association is represented by assigning surface coordinates to points on the mesh that lie within the parallel region, augmenting the

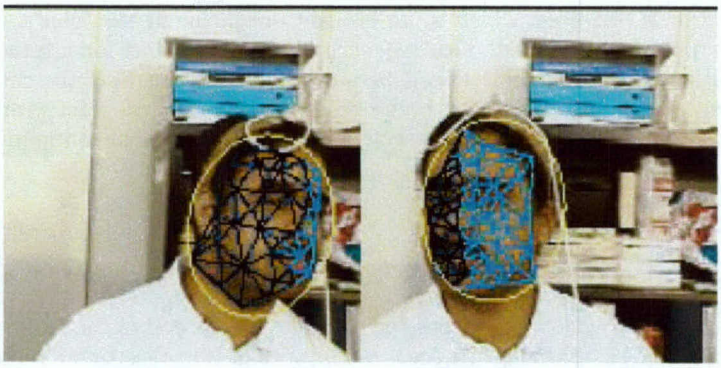

Figure 10: A visualization of surface coordinates on the mesh. The mesh has been colored here based on the sign of a surface coordinate, so that it appears as two halves locked onto either side of the face.

image plane coordinates they jointly possess. As the parallel region shifts during a rotation in depth, new points entering the region are assigned surface coordinates based on their image plane coordinates, with the transformation between the two easy to maintain using the rotation, scaling, and translation of the mesh already recovered.

Independently of the argument given earlier for the types of movements that can be tracked without accurate knowledge of the shape of the head, the mesh allows a new set of trajectories to be tracked: those which leave some portion of the face visible throughout. The surface coordinates of points on the mesh covering that part of the face can be used as landmarks.

3D pose recovery

Recovery of the 3D location of the head is straightforward, given knowledge of the camera's parameters, although there is of course a scale/depth ambiguity since no absolute depth information is recovered.

Recovery of 3D orientation is equally straightforward, but shape dependent. The output of the tracker is effectively a procedure for turning a specified point on the surface of the object towards the camera and then rotating it to a specified degree. To convert this into Euler angles, for example, requires knowledge of the shape of the object so that surface points can be associated with vectors from wherever the center of the head is taken to be. At this point, we must make use of the estimates for the dimensions of the head from the head tracker and make the conversion using a simple ellipsoidal model. The crucial point is that inaccuracies in this process do not feed back to the tracker itself.

Current results

The system has so far been tested on a data-set made available by Sclaroff et al (Cascia, Sclaroff & Athitsos 2000), consisting of video of head movements with ground truth measured by a Flock of Birds sensor on the subjects' heads. These sequences are 200 frames in duration. To test the stability of the tracker over long intervals, the Sclaroff sequences are here artificially extending by looped them forward and back for twenty iterations. Figure 11 shows tracking results for the sequence which appeared to have the largest rotation in depth (in no case unfortunately did the eyes become occluded, which would have made for a better demonstration of the advantages of the system developed in this paper). Angular measurements are limited by the accuracy with which they can be initialized, which turns out to be to within about 5° for roll and yaw, and about 10° for pitch. Because of re-initialization events, estimates of pose will contain

discontinuities when drift is corrected, which is not brought out in the figure. This could be dealt with for estimation of pose across a pre-recorded video sequence like this one, but for use in a vision interface it seems the discontinuities are unavoidable. This is because the best estimate of the current pose does truly change instantaneously when an initialization even occurs, and there is no point propagating information backwards to previous frames during real-time interaction unless there is some background processing going on that can have high latency.

Alternative approaches

There are many possible approaches to head pose estimation. At one end of the scale, there are techniques that rely on very strong models of head shape, such as the work of Horprasert and Yacoob (Black & Yacoob 1995). The shape of the human head is broadly similar across the species. Anthropometry characterizes the distribution of face length scales and ratios within different sub-groups. These distributions are quite narrow for a subject whose gender, race, and age are known. Horprasert and Yacoob make use of this to estimate head orientation from monocular images. They show that pose can be recovered by tracking just five points on the face (four at the eye corners and a fifth at the tip of the nose), given that the necessary anthropometric data is available. They propose a two stage system that estimates a subjects gender, race and age first, indexes into the appropriate table of anthropometric data, and then performs the pose estimation.

At the other end of the scale, there are pose tracking systems which do not require a prior model, and are therefore of more general application than systems that rely on special characteristics of the head – for example Harville et al (Harville, Rahimi, Darrell, Gordon & Woodfill 1999).

Other points on the spectrum include the application of eigenspace techniques to directly recognize the pose of a specific user, as opposed to tracking changes in pose (McKenna & Gong 1998). And then there are very many systems designed to run in real-time, using a wide variety of simple cues such as hair outline (Wang & Brandstein 1998).

9 Motor control

This thesis work will be implemented on the robot Cog, an upper torso humanoid (Brooks et al. 1999). The robot's mechanical degrees of freedom can be organized into three groups: the torso, the head, and the arms, as illustrated in Figure 12. The robot has previously been applied to tasks such as visually-guided pointing (Marjanović et al. 1996), and rhythmic operations such as turning a crank or driving a slinky (Williamson 1998a). This section describes how motor control has been implemented, highlighting changes needed for this work.

9.1 The arms

Cog has two arms, each of which has six degrees of freedom organized as shown in Figure 13. The joints are driven by series elastic actuators (Williamson 1995) – essentially a motor connected to its load via a spring (think strong and torsional rather than loosely coiled). The arm is not designed to enact trajectories with high fidelity. For that a very stiff arm is preferable. Rather, it is designed to perform well when interacting with a poorly characterized environment. The spring acts as a low pass filter for the friction and backlash effects introduced by gears, and protects the gear teeth from shearing under the impact of shock loads.

A drawback to the use of series elastic actuators is that they limit the control bandwidth in cases where the applied force needs to change rapidly. The force applied by an electric motor can normally be changed rapidly, since it is directly proportional to the current supplied. By putting a motor in series with a spring,

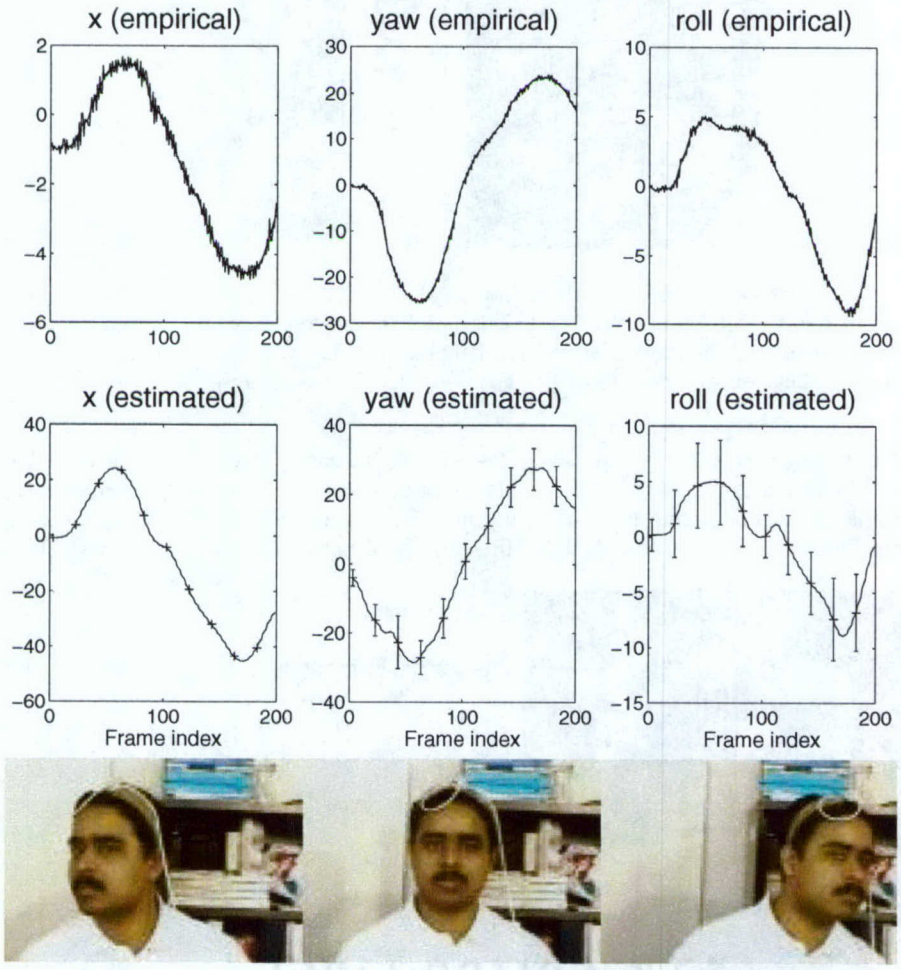

Figure 11: Results for a sequence containing a yaw movement and horizontal translation, with all other parameters remaining basically unchanged except for a slight roll. The top row shows ground truth. The second row shows the estimated pose parameters that change significantly during the sequence. The estimated x coordinate is left in terms of the image plane. Values plotted are averaged for each occurrence of a particular frame over a *single tracking run* constructed from a sequence being played, then played in reverse, then repeated again for twenty iterations. Error bars show the standard deviation of estimates for each frame. There is about a 5° error in angles, which in this case means the roll estimate is mostly noise.

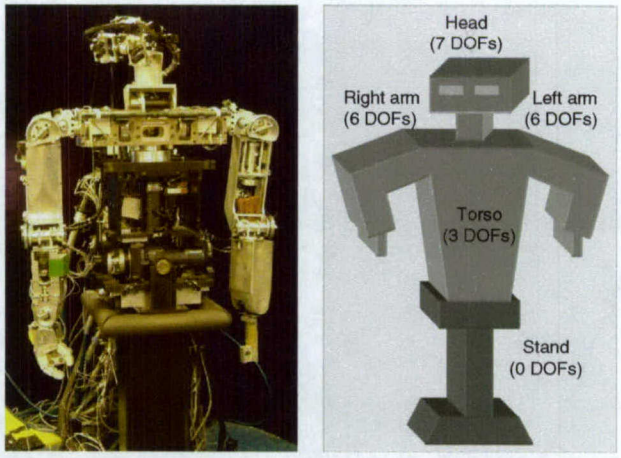

Figure 12: Degrees of freedom (DOFs) of the robot Cog. The arms terminate either in a primitive "flipper" or a four-fingered hand (seen on the far left). The hand is an independent research project by Matthew Marjanović, and will not be employed or described here. The head, torso, and arms together contain 22 degrees of freedom. The robot is on a fixed platform – it does not locomote.

this ability is lost, since the motor must now drive a displacement of the spring's mass before the applied force changes. This is not anticipated to be a problem for this research, and has proven an acceptable trade-off in some simple tests. The actuators controlling the robot's head are not coupled with springs, since they unarguably need high bandwidth, and the head ideally should not come into contact with the environment.

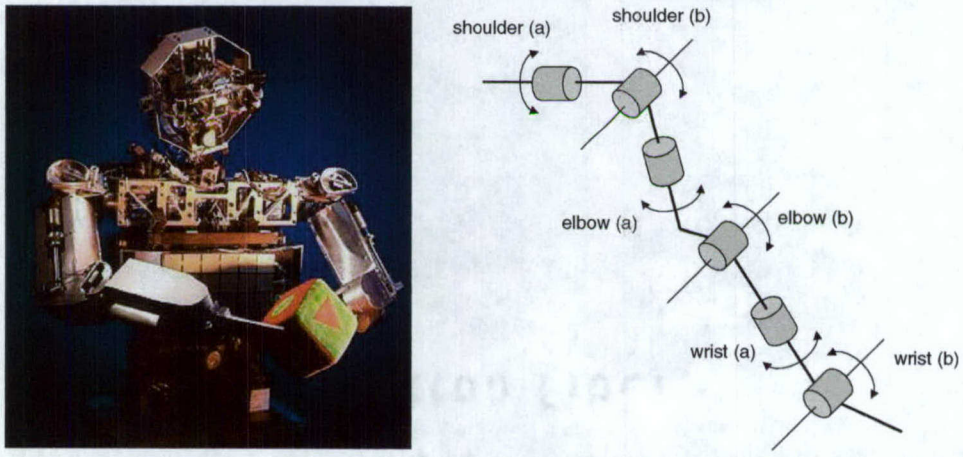

Figure 13: Kinematics of the arm, following Williamson (1999). There are a total of six joints, divided into a pair for each of the shoulder, elbow, and wrist/flipper.

Low-level arm control

The arms are driven by two nested controllers. The first implements force control, driving the motor until a desired deflection of the spring is achieved, as measured by a strain gauge. This control loop is implemented using an 8-axis motor controller from Motion Engineering, Inc. A second loop controls the deflection setpoint to achieve a desired joint angle as measured by a potentiometer. Figure 14 shows this second loop as it has

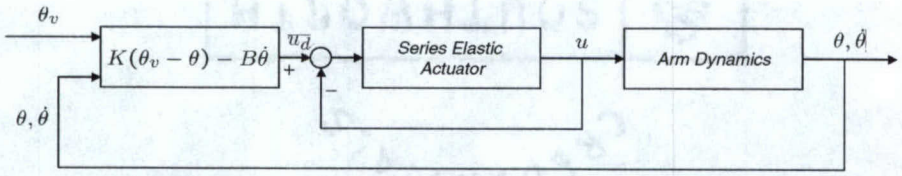

Figure 14: Control of a joint in the arm, following Williamson (1999). An inner loop controls the series elastic actuator in terms of force, working to achieve a desired deflection of the spring as measured by a strain gauge. An outer loop controls the deflection setpoint to achieve a desired joint angle, as measured by a potentiometer.

been implemented for Cog in the past, using a spring/damping control law (Williamson 1999). This was appropriate for rhythmic, resonant tasks. For this thesis work, a simple PID control law will be used instead, but the basic control architecture is the same. Giorgio Metta, who is working on mirror neuron theories with Cog, has been updating Cog's motor control after some recent changes to the arm, and I am currently using his controller.

Tactics and strategy for arm control

Robot arms are usually employed for the purposes of manipulation, but for this work they instead serve primarily as aides to the visual system. The target of a reaching operation is not assumed to be well characterized; in fact the reaching operation serves to better define the characteristics of the target through active segmentation (Figure 15). Hence the arm will habitually be colliding with objects. Sometimes the collisions will be with rigid, more or less unyielding structures such as a table. Sometimes the collisions will be with movable objects the robot could potentially manipulate. And sometimes the collisions will be with people. A reaching strategy which provides useful information to the visual system while avoiding damage to the arm or the environment will be an important part of this work. Qualitatively, having batted Cog's arms around and suffering being accidentally prodded by them on occasion, the physical properties of the arm are well suited to this work. An important part of the perceptual component of this thesis will be developing good strategies for poking objects that are initially poorly localized both in terms of 3D location and in terms of extent.

9.2 The head

Figure 16 shows the degrees of freedom associated with Cog's head. In each "eye", a pair of cameras with different fields of view provides a step-wise approximation to the smoothly varying resolution of the human fovea (Scassellati 1998). The eyes pan independently and tilt together. The head rolls and tilts through a differential drive. There is a further pan and tilt associated with the neck. There are a number of redundancies in the degrees of freedom to permit rapid movement of the eyes followed by a slower compensating motion of the relatively massive head. The head contains a 3-axis inertial sensor to simplify gaze stabilization.

Head control

The motors of the head are connected to optical encoders and driven by an 8-axis motor controller from Motion Engineering, Inc. The motor controller is configured to permit both position and velocity control. Much has been written about both the low-level and strategic control of such a head (Scassellati 2001) and this will not play a significant part of this thesis. I intend to use a variant of a controller developed for a similar head (Breazeal et al. 2000).

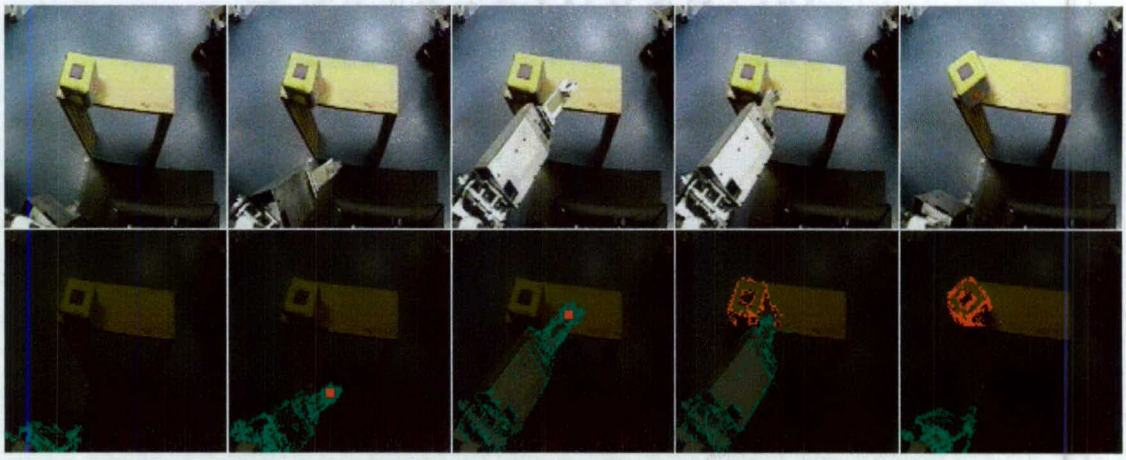

Figure 15: A motivating example of one of Cog's arms in operation. The upper sequence shows an arm extending into a workspace, tapping an object, and retracting. This is an exploratory mechanism for finding the boundaries of objects, and essentially requires the arm to collide with objects under normal operation, rather than as an occasional accident. The lower sequence shows in red the shape identified from the tap using simple image differencing and flipper tracking.

9.3 The torso

Cog can bend forwards and from side-to-side around axes roughly corresponding to the human torso, as shown in Figure 17. The torso can also twist about its base. The torso motors are controlled in a manner similar to the arm motors. While they are useful for accessing a wider range of workspaces, they will most likely not be used for this thesis.

10 Dialog management

Communicating a task to the robot will make use of verbal interaction across an extended period of time. While this component of the interaction is not the focus of the thesis, it is appropriate to address some of the theoretic and practical issues that arise in lengthy dialogues.

10.1 Deixis and Anaphora

This thesis work rides roughshod over accepted linguistic categories, particularly a distinction drawn between deictic and anaphoric references. Both types of references are context-dependent, and are distinguished by whether their meaning is a function of entities external to the discourse (deictic reference) or introduced within the discourse (anaphoric reference). For example, pronouns can be divided in this way:

> ... deictic and anaphoric pronouns select their referents from certain sets of antecedently available entities. The two pronoun uses differ with regard to the nature of these sets. In the case of a deictic pronoun the set contains entities that belong to the real world, whereas the selection set for an anaphoric pronoun is made up of constituents of the representation that has been constructed in response to antecedent discourse.
>
> Kamp (1981)

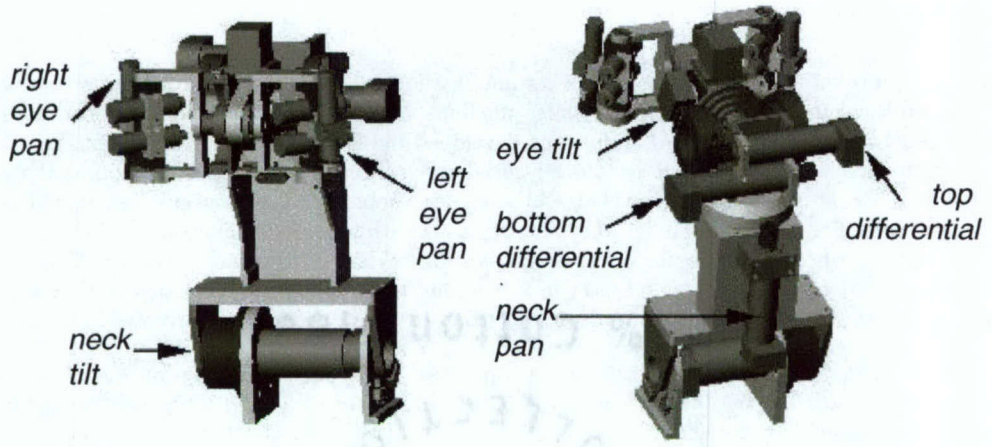

Figure 16: The motors of Cog's head, following Scassellati (2001). The degrees of freedom are loosely organized as pertaining to either the eyes, head, or neck. Pan and tilt (but not roll) of the eyes can be achieved at high speed without moving the mass of the head.

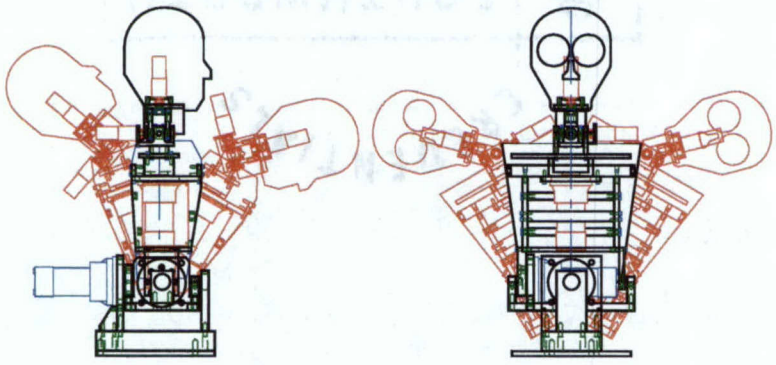

Figure 17: The torso, following Brooks et al. (1999). The body can lean forward, backwards, and to the sides. It can also rotate about its base (not shown).

While this may be a useful analytic distinction to make when characterizing human linguistic behavior, it seems less fruitful for the purposes of synthesis. Dividing referents into entities that belong to the "real world" and representational entities that have been constructed through discourse does not seem useful. The real world no doubt exists, and certainly it can be carved up into entities in a variety of ways, but the robot has only weak access to such entities through perception. In a strong sense those "real world" entities do not exist to the robot except as seen through the filters of the representations constructed through discourse, as described in Section 3.

The ability to refer to entities introduced during discourse is in a sense what this thesis is all about. But the manner in which the entities are introduced and the means by which they are referred to is much more indirect than usually envisaged in the linguistics literature. The kinds of tasks that can in practice be communicated will be limited by this indirectness and sublinguistic nature, and the thesis will not address classical linguistic anaphora. The hope is that experience with mechanisms that establish reference through extended interactions could provide a more robust toolbox with which to approach successively shorter and more stylized means of reference.

10.2 Speech errors

Speech recognition is subject to error, so verbal communication will require procedures for error correction, just like any protocol implemented across a noisy medium. Preferably this should not lead to excessive verbosity or repetition. A useful strategy for this developed by the dialog systems community is *implicit verification* (Souvignier et al. 2000). Here information derived from user speech is incorporated within the system's own speech output in a manner that will not excite comment if it is accurate but will otherwise prompt a correction. For example, if a travel system recognized the user's destination as "Austin", then this could be implicitly verified by expanding a follow-up question "When do you want to depart?" to be "When do you want to depart for Austin?". If the user in fact wishes to go to Boston, not Austin, they will now be aware that the system is laboring under a misapprehension that will need to be corrected.

In our scenario, when the human is demonstrating a process, the robot can simply repeat what it thought it heard. If this is incorrect, the human can tag this verbally as an error and retry, otherwise the session simply proceeds without any further comment. Of course, other classes of error are now possible through confusion of the error tag with normal input, and corrections themselves are notoriously error-prone (Swerts, Hirschberg & Litman 2000). Such errors can be dealt with if they are sufficiently infrequent; if not, then it is better to back off to a more conservative verification strategy. Initial tests suggest that error rates will be low enough to permit implicit verification (although motor noise is a problem).

If errors are frequent, it is desirable that dialog be as stateless as possible to avoid confusion about dialog state. While the task communication procedure described in Section 3 needs to be stateful, with an N-gram representation it is trivial to drop the immediate history of a process when an error occurs while still retaining most of the structure already communicated. In simulated trials, this seems much simpler than trying to bring the robot and user back into agreement about that history.

11 Conclusions

We can pose the problem of tasking as essentially one of communicating the contextual information needed to make the perceptual and control problems well-posed. We are parameterizing across sets of constraints, rather than eliminating them. Rather than requiring immediate solutions to general, hard, problems such as object segmentation and motion understanding, we focus on communicative protocols to enable the robot to solve particular problems inherent in a given task, such as detecting the difference between two types of widgets that need to be sorted (for example).

12 Timeline

Some of the components of this system have been implemented for Kismet, an expressive robot head. The full system will be targeted for the Cog system – a robot head, torso, and arms.

Date		Milestone
September	2001	Complete prototype task modeling module
October	2001	Gather robot's percepts into a well-defined network
November	2001	Translate task model and demonstration into examples and labels for ML
December	2002	Implement basic task grounding module to solve ML tasks (without external aid in feature selection)
January	2002	(Area Exam)
February	2002	Upgrade speech processing to facilitate feature binding
March	2002	Implement feature selection protocol
April	2002	Demonstrate grounding of a simple sorting task
May	2002	Implement object poking (active segmentation)
June	2002	Demonstrate a sorting task requiring active measurements
July	2002	Upgrade task modeling to detect task/episode boundaries
August	2002	Upgrade speech processing to facilitate task reference and parameterization
September	2002	Demonstrate rapid invocation of a sorting task
October	2002	Implement spatial memory for reference
November	2002	Demonstrate mixed vocal, gestural invocation of a task
December	2002	Implement explicit role slots for processes
January	2003	Demonstrate comprehension of a searching task with absent referents
February	2003	Implement invocation by goal rather than by method
March	2003	Demonstrate novel active measurements introduced as tasks and used as goals
April	2003	Solve iterated problems with cascading reference

References

Adams, B., Breazeal, C., Brooks, R. & Scassellati, B. (2000). The Cog project, *IEEE Intelligent Systems* . To appear.

Aslin, R., Woodward, J., LaMendola, N. & Bever, T. (1996). Models of word segmentation in fluent maternal speech to infants, *in* J. Morgan & K. Demuth (eds), *Signal to Syntax: Bootstrapping From Speech to Grammar in Early Acquisition*, Lawrence Erlbaum Associates: Mahwah, NJ.

Ballard, D. (1989). Behavioral constraints on animate vision, *Image and Vision Computing* **7:1**: 3–9.

Bard, E. & Anderson, A. (1994). The unintelligibility of speech to children: effects of referent availability, *Journal of Child Language* **21**: 623–648.

Basu, S., Essa, I. & Pentland, A. (1996). Motion regularization for model-based head tracking, *Intl. Conf. on Pattern Recognition*, Vienna, Austria.

Bazzi, I. & Glass, J. (2000). Modeling out-of-vocabulary words for robust speech recognition, *Proc. 6th International Conference on Spoken Language Processing*, Beijing, China.

Beardsley, P. A. (1998). A qualitative approach to classifying head and eye pose, *IEEE Workshop on Applications of Computer Vision*, Florence, Italy, pp. 208–213.

Billard, A. (2001). Imitation: a means to enhance learning of a synthetic proto-language in an autonomous robot, *in* K. Dautenhahn & C. L. Nehaniv (eds), *Imitation in Animals and Artifacts*, MIT Press.

Billard, A. & Dautenhahn, K. (1997). Grounding communication in situated, social robots, *Technical report*, University of Manchester.

Birchfield, S. (1998). Elliptical head tracking using intensity gradients and color histograms, *CVPR*, pp. 232–237.

Black, M. & Yacoob, Y. (1995). Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion, *ICCV*, pp. 374–381.

Bloom, P. (2000). *How Children Learn the Meaning of Words*, Cambridge: MIT Press.

Blumberg, B. (1996). *Old Tricks, New Dogs: Ethology and Interactive Creatures*, PhD thesis, MIT.

Borchardt, G. C. (1993). Causal reconstruction, *Technical Report AIM-1403*, MIT Artificial Intelligence Laboratory.

Breazeal, C. (2000). *Sociable Machines: Expressive Social Exchange Between Humans and Robots*, PhD thesis, MIT Department of Electrical Engineering and Computer Science.

Breazeal, C. & Aryananda, L. (2000). Recognition of affective communicative intent in robot-directed speech, *Proceedings of Humanoids 2000*, Cambridge, MA.

Breazeal, C., Edsinger, A., Fitzpatrick, P., Scassellati, B. & Varchavskaia, P. (2000). Social constraints on animate vision, *IEEE Intelligent Systems* **15**.

Breazeal, C. & Scassellati, B. (2000). Infant-like social interactions between a robot and a human caretaker, *Adaptive Behavior* **8**(1). To appear.

Breazeal, C. & Velasquez, J. (1998). Toward teaching a robot "infant" using emotive communication acts, *Socially Situated Intelligence: Papers from the 1998 Simulated Adaptive Behavior Workshop.*

Brent, M. & Siskind, J. (2001). The role of exposure to isolated words in early vocabulary development, *Cognition* **81**: B33–B44.

Brooks, R. A. (1986). A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation* **RA-2**: 14–23.

Brooks, R. A. (1991a). Intelligence without reason, *Proceedings of the 1991 International Joint Conference on Artificial Intelligence*, pp. 569–595.

Brooks, R. A. (1991b). Intelligence without representation, *Artificial Intelligence Journal* **47**: 139–160. originally appeared as MIT AI Memo 899 in May 1986.

Brooks, R. A., Breazeal, C., Irie, R., Kemp, C. C., Marjanović, M., Scassellati, B. & Williamson, M. M. (1998). Alternative essences of intelligence, *Proceedings of the American Association of Artificial Intelligence (AAAI-98).*

Brooks, R. A., Breazeal, C., Marjanovic, M. & Scassellati, B. (1999). The Cog project: Building a humanoid robot, *Lecture Notes in Computer Science* **1562**: 52–87.

Brooks, R. A. & Stein, L. A. (1994). Building brains for bodies, *Autonomous Robots* **1**(1): 7–25.

Bullowa, M. (1979). *Before Speech: The Beginning of Interpersonal Communicaion*, Cambridge University Press, Cambridge, London.

Burnham, D., Francis, E., Kitamura, C., Vollmer-Conna, U., Averkiou, V., Olley, A. & Paterson, C. (1998). Are you my little pussy-cat? acoustic, phonetic and affective qualities of infant- and pet-directed speech, *Proc. 5th International Conference on Spoken Language Processing*, Vol. 2, pp. 453–456.

Butterworth, G. (1991). The ontogeny and phylogeny of joint visual attention, *in* A. Whiten (ed.), *Natural Theories of Mind*, Blackwell.

Cascia, M. L., Sclaroff, S. & Athitsos, V. (2000). Fast, reliable head tracking under varying illumination: An approach based on robust registration of texture-mapped 3D models, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 22(4).

Cassell, J. (1989). Embodied conversation: integrating face and gesture into automatic spoken dialogue systems.

Cassell, J., Pelachaud, C., Badler, N., Steedman, M., Achorn, B., Becket, T., Douville, B., Prevost, S. & Stone, M. (1994). Animated conversation: Rule-based generation of facial expression, gesture, and spoken intonation for multiple conversational agents, *SIGGRAPH'94* .

Chapman, D. & Agre, P. E. (1987). Pengi: An implementation of a theory of activity, *Proceedings of the Sixth National Conference on Artificial Intelligence*, pp. 268–272.

Chen, Q., amd T. Shioyama, H. W. & Shimada, T. (1999). 3D head pose estimation using color information, *6th IEEE International Conference on Multimedia Computing and Systems*, Florence, Italy.

Colombetti, M. & Dorigo, M. (1994). Training agents to perform sequential behavior, *Adaptive Behavior* **2**(3).

Connell, J. (1989). A colony architecture for an artificial creature, *Technical Report AITR-1151*, Massachusetts Institute of Technology.

Cordea, M., Petriu, E., Georganas, N., Petriu, D. & Whalen, T. (2000). Real-time 2.5D head pose recovery for model-based video-coding, *IEEE Instrumentation and Measurement Technology Conference*, Baltimore, MD, USA.

DeCarlo, D. & Metaxas, D. (1996). The integration of optical flow and deformable models with applications to human face shape and motion estimation, *CVPR*, pp. 231–238.

Dennett, D. C. (1987). *The Intentional Stance*, MIT Press.

Duda, R. & Hart, P. (1973). *Pattern Classification and Scene Analysis*, Wiley, New York.

Ferrell, C. (1998). Learning by scaffolding. MIT Ph.D. Thesis Proposal.

Ferrell, C. & Kemp, C. (1996). An ontogenetic perspective to scaling sensorimotor intelligence, *Embodied Cognition and Action: Papers from the 1996 AAAI Fall Symposium*, AAAI Press.

Firby, R. J. (1994). Task networks for controlling continuous processes, *Proceedings of the Second International Conference on AI Planning Systems*, Chicago IL.

Garland, A. & Lesh, N. (2001). Learning hierarchical task models by demonstration, *Technical report*, Mitsubishi Electric Research Laboratories.

Gat, E. (1996). ESL: A language for supporting robust plan execution in embedded autonomous agents, *Plan Execution: Problems and Issues: Papers from the 1996 AAAI Fall Symposium*, AAAI Press, Menlo Park, California, pp. 59–64.

Glass, J., Chang, J. & McCandless, M. (1996). A probabilistic framework for feature-based speech recognition, *Proc. International Conference on Spoken Language Processing*, pp. 2277–2280.

Glass, J. & Weinstein, E. (2001). Speechbuilder: Facilitating spoken dialogue systems development, *7th European Conference on Speech Communication and Technology*, Aalborg, Denmark.

Goldberg, D. & Mataric, M. J. (1999). Augmented markov models, *Technical report*, USC Institute for Robotics and Intelligent Systems.

Goldberg, M. E. (2000). The control of gaze, *in* E. R. Kandel, J. H. Schwartz & T. M. Jessell (eds), *Principles of Neural Science*, 4rd edn, McGraw-Hill.

Goodman, N. (1983). *Fact, Fiction, and Forecast*, Harvard University Press, Cambridge, Massachusetts.

Gorin, A., Petrovksa-Delacrtaz, D., Riccardi, G. & Wright, J. (1999). Learning spoken language without transcriptions, *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, Colorado.

Grice, H. (1974). Logic and conversation, *in* P. Cole & J. Morgan (eds), *Syntax and semantics*, Vol. 3, Academic Press, New York, pp. 41–58.

Haigh, K. Z. & Veloso, M. M. (1998). Planning, execution and learning in a robotic agent, *AIPS-98*, pp. 120 – 127.

Halliday, M. (1975). *Learning How to Mean: Explorations in the Development of Language*, Elsevier, New York, NY.

Harville, M., Rahimi, A., Darrell, T., Gordon, G. & Woodfill, J. (1999). 3D pose tracking with linear depth and brightness constraints, *ICCV*, pp. 206–213.

Hauser, M. D. (1996). *Evolution of Communication*, MIT Press.

Hazen, T. & Bazzi, I. (2001). A comparison and combination of methods for OOV word detection and word confidence scoring, *Proc. International Conference on Acoustics*, Salt Lake City, Utah.

Heinzmann, J. & Zelinsky, A. (1997). Robust real-time face tracking and gesture recognition, *Proc. International Joint Conference on Artificial Intelligence*, Vol. 2, pp. 1525–1530.

Hirschberg, J., Litman, D. & Swerts, M. (1999). Prosodic cues to recognition errors, *ASRU*.

Horn, B. K. P. (1986). *Robot Vision*, MIT Press.

Horprasert, T., Yacoob, Y. & Davis, L. S. (1997). An anthropometric shape model for estimating head orientation, *3rd International Workshop on Visual Form*, Capri, Italy.

Horswill, I. (1993). *Specialization of Perceptual Processes*, PhD thesis, MIT.

Jeanrenaud, P., Ng, K., Siu, M., Rohlicek, J. & Gish, H. (1993). Phonetic-based word spotter: Various configurations and application to event spotting, *Proc. EUROSPEECH*.

Jusczyk, P. (1997). *The Discovery of Spoken Language*, Cambridge: MIT Press.

Jusczyk, P. & Aslin, R. (1995). Infants' detection of the sound patterns of words in fluent speech, *Cognitive Psychology* **29**: 1–23.

Kaelbling, L. P., Littman, M. L. & Moore, A. P. (1996). Reinforcement learning: A survey, *Journal of Artificial Intelligence Research* **4**: 237–285.

Kaelbling, L. P., Oates, T., Hernandez, N. & Finney, S. (2001). Learning in worlds with objects, *AAAI Spring Symposium*.

Kamp, H. (1981). A theory of truth and semantic representation, *in* G. J, T. Janssen & M. Stokhof (eds), *Formal Methods in the Study of Language*, Mathematical Center Tract 135, Amsterdam, pp. 277–322.

Klingspor, V., Demiris, J. & Kaiser, M. (1997). Human-robot-communication and machine learning, *Applied Artificial Intelligence Journal* **11**: 719–746.

Koga, Y., Kondo, K., Kuffner, J. & Latombe, J.-C. (1994). Planning motions with intentions, *Computer Graphics* **28**(Annual Conference Series): 395–408.

Kozima, H. (1998). Attention-sharing and behavior-sharing in human-robot communication, *IEEE International Workshop on Robot and Human Communication (ROMAN-98, Takamatsu)*, pp. 9–14.

Lakoff, G. (1987). *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*, University of Chicago Press, Chicago, Illinois.

Lenat, D. B. (1995). CYC: A large-scale investment in knowledge infrastructure, *Communications of the ACM* **38**(11): 33–38.

Marjanovic, M. (1995). *Learning functional maps between sensorimotor systems on a humanoid robot*, Master's thesis, MIT Department of Electrical Engineering and Computer Science.

Marjanović, M. J., Scassellati, B. & Williamson, M. M. (1996). Self-taught visually-guided pointing for a humanoid robot, *From Animals to Animats: Proceedings of 1996 Society of Adaptive Behavior*, Cape Cod, Massachusetts, pp. 35–44.

Markman, E. M. (1989). *Categorization and naming in children: problems of induction*, MIT Press, Cambridge, Massachusetts.

Mataric, M. J. (1990). A distributed model for mobile robot environment-learning and navigation, *Technical Report AITR-1228*, Massachusetts Institute of Technology.

Mataric, M. J. (2000). Getting humanoids to move and imitate, *IEEE Intelligent Systems* pp. 18–24.

Matsusaka, Y. & Kobayashi, T. (1999). Human interface of humanoid robot realizing group communication in real space, *Proc. Second International Symposium on Humanoid Robots*, pp. 188–193.

McCarthy, J. & Hayes, P. J. (1969). Some philosophical problems from the standpoint of artificial intelligence, *in* B. Meltzer & D. Michie (eds), *Machine Intelligence 4*, Edinburgh University Press, pp. 463–502. reprinted in McC90.

McKenna, S. & Gong, S. (1998). Real-time face pose estimation, *International Journal on Real Time Imaging, Special Issue on Real-time Visual Monitoring and Inspection.* **4**: 333–347.

Minsky, M. (1985). *The Society of Mind*, Simon and Schuster, New York.

Minsky, M. (1990). Logical vs. analogical or symbolic vs. connectionist or neat vs. scruffy, *in* P. H. Winston (ed.), *Artificial Intelligence at MIT., Expanding Frontiers*, Vol. 1, MIT Press. Reprinted in AI Magazine, 1991.

Mitchell, T. (1980). The need for biases in learning generalizations, *Technical report*, Computer Science Department, Rutgers University.

Münch, S., Kreuziger, J., Kaiser, M. & Dillmann, R. (1994). Robot programming by demonstration (RPD) – using machine learning and user interaction methods for the development of easy and comfortable robot programming systems, *24th International Symposium on Industrial Robots.*

Nicolescu, M. & Mataric, M. J. (2001). Experience-based learning of task representations from human-robot interaction, *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Alberta, Canada, pp. 463–468.

Oates, T. (1999). Identifying distinctive subsequences in multivariate time series by clustering, *Knowledge Discovery and Data Mining*, pp. 322–326.

Oates, T., Eyler-Walker, Z. & Cohen, P. (2000). Toward natural language interfaces for robotic agents: Grounding linguistic meaning in sensors, *Proceedings of the 4th International Conference on Autonomous Agents*, pp. 227–228.

Oates, T., Jensen, D. & Cohen, P. (1998). Discovering rules for clustering and predicting asynchronous events, *Predicting the Future: AI Approaches to Time-Series Problems*, AAAI Press, pp. 73–79.

Pepperberg, I. (1990). Referential mapping: A technique for attaching functional significance to the innovative utterances of an african grey parrot, *Applied Psycholinguistics* **11**: 23–44.

Quine, W. V. O. (1960). *Word and object*, Harvard University Press, Cambridge, Massachusetts.

Roy, D. (1999). *Learning Words from Sights and Sounds: A Computational Model*, PhD thesis, MIT.

Scassellati, B. (1998). A binocular, foveated active vision system, *Technical Report 1628*, MIT Artificial Intelligence Lab Memo.

Scassellati, B. (1999). Imitation and mechanisms of joint attention: A developmental structure for building social skills on a humanoid robot, *in* C. L. Nehaniv (ed.), *Computation for Metaphors, Analogy and Agents*, Vol. 1562 of *Springer Lecture Notes in Artificial Intelligence*, Springer-Verlag.

Scassellati, B. (2000). Theory of mind for a humanoid robot, *Proceedings of the First International IEEE/RSJ Conference on Humanoid Robotics.*

Scassellati, B. (2001). *Foundations for a Theory of Mind for a Humanoid Robot*, PhD thesis, MIT Department of Electrical Engineering and Computer Science.

Sherrah, J. & Gong, S. (2000). Fusion of perceptual cues for robust tracking of head pose and position, *to appear in Pattern Recognition.*

Shi, J. & Tomasi, C. (1994). Good features to track, *CVPR*, pp. 593 – 600.

Sigal, L., Sclaroff, S. & Athitsos, V. (2000). Estimation and prediction of evolving color distributions for skin segmentation under varying illumination, *CVPR*, Vol. 2, pp. 152–159.

Smith-Mickelson, J. (2000). *Design and application of a head detection and tracking system*, Master's thesis, MIT.

Souvignier, B., Kellner, A., Rueber, B., Schramm, H. & Seide, F. (2000). The thoughtful elephant: strategies for spoken dialog systems, *IEEE Transactions on Speech and Audio Processing* **8**(1): 51–62.

Stauffer, C. & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking, *CVPR99.*

Steels, L. (1996). Emergent adaptive lexicons, *Proceedings of the fourth international conference on simulation of adaptive behavior.*

Strom, J., Jebara, T., Basu, S. & Pentland, A. (1999). Real time tracking and modeling of faces: An EKF-based analysis by synthesis approach, *Modelling People Workshop, ICCV.*

Swerts, M., Hirschberg, J. & Litman, D. (2000). Corrections in spoken dialogue systems, *Proceedings of the International Conference on Spoken Language Processing*, Beijing, China.

Thrun, S. (1998). A framework for programming embedded systems: Initial design and results, *Technical Report CMU-CS-98-142*, Carnegie Mellon University.

Tomasello, M. (1997). The pragmatics of word learning, *Japanese Journal of Cognitive Science* .

Trevarthen, C. (1979). Communication and cooperation in early infancy: a description of primary intersubjectivity, *in* M. Bullowa (ed.), *Before Speech: The beginning of interpersonal communication*, Cambridge University Press.

Ullman, S. (1984). Visual routines, *Cognition* **18**: 97–159. (Also in: Visual Cognition, S. Pinker ed., 1985).

Varchavskaia, P. & Fitzpatrick, P. (2001). Characterizing and processing robot-directed speech, *submitted to Humanoids 2001*, Tokyo, Japan.

Vygotsky, L. (1962). *Thought and language*, MIT Press, Cambridge, MA.

Wang, C. & Brandstein, M. (1998). A hybrid real time face tracking system, *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing.*, Seattle.

Wang, X. (1996). Planning while learning operators, *in* B. Drabble (ed.), *Proceedings of the 3rd International Conference on Artificial Intelligence Planning Systems (AIPS-96)*, AAAI Press, pp. 229–236.

Werker, J., Lloyd, V., Pegg, J. & Polka, L. (1996). Putting the baby in the bootstraps: Toward a more complete understanding of the role of the input in infant speech processing, *in* J. Morgan & K. Demuth (eds), *Signal to Syntax: Bootstrapping From Speech to Grammar in Early Acquisition*, Lawrence Erlbaum Associates: Mahwah, NJ, pp. 427–447.

Williamson, M. (1995). *Series elastic actuators*, Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Williamson, M. (1998a). Neural control of rhythmic arm movements, *Neural Networks* **11**(7-8): 1379–1394.

Williamson, M. (1999). *Robot Arm Control Exploiting Natural Dynamics*, PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Williamson, M. M. (1996). Postural primitives: Interactive behavior for a humanoid robot arm, *Fourth International Conference on Simulation of Adaptive Behavior*, Cape Cod, Massachusetts, pp. 124–131.

Williamson, M. M. (1998b). Exploiting natural dynamics in robot control, *Fourteenth European Meeting on Cybernetics and Systems Research (EMCSR '98)*, Vienna, Austria.

Winston, P. (1970). *The Psychology of Computer Vision*, McGraw-Hill, New York, chapter Learning Structure Descriptions from Examples, pp. 157–209.

Wren, C., Azarbayejani, A., Darrell, T. & Pentland, P. (1997). Pfinder: Real-time tracking of the human body, *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

Y. Wu, K. T. & Huang, T. S. (2000). Wide-range, person- and illumination-insensitive head orientation estimation, *Proc. of Int'l Conf. on Face and Gesture Recognition*, Grenoble, France.

Yip, K. & Sussman, G. J. (1997). Sparse representations for fast, one-shot learning, *Proc. of National Conference on Artificial Intelligence*.

Zue, V. & Glass, J. (2000). Conversational interfaces: Advances and challenges, *Proceedings of the IEEE, Special Issue on Spoken Language Processing* **Vol. 88**.

Zue, V., Glass, J., Plifroni, J., Pao, C. & Hazen, T. (2000). Jupiter: A telephone-based conversation interface for weather information, *IEEE Transactions on Speech and Audio Processing* **8**: 100–112.